messen steuern regeln

msr



VEB Verlag Technik Berlin
EVP: 4,— M
ISSN 0026-0347

6/89



messen ■ steuern ■ regeln

Wissenschaftlich-technische Zeitschrift für die Automatisierungstechnik

Herausgeber:

Wissenschaftlich-Technische Gesellschaft für Meß- und Automatisierungstechnik (WGMA) in der KDT

Zum Titelbild:

Durch energieoptimale Führung von Trocknungsprozessen können erhebliche Energiemengen eingespart werden. Unser Titelbild zeigt eine dazu erforderliche Abluftfeuchteregelanlage. Wesentlicher Bestandteil ist ein geeigneter Abluftfeuchtesensor, der den extremen Forderungen der industriellen Praxis — heiße und verschmutzte Abluft — genügt. Einen neuen Gasfeuchtesensor, der diesen Bedingungen gerecht wird, stellen wir auf S. 268 vor.

Titelgrafik: G. SCHWESINGER

6

1989

32. Jahrgang
Juni 1989

ISSN 0026-0347

VEB
VERLAG TECHNIK
BERLIN

Aufsätze

M. Hartwig, E. Stein, R. Strobel

Programmieren mit Ada

0. Einleitung

Die Sprache Ada[1] entstand Ende der 70er Jahre als ein Versuch, die zum Teil anachronistischen, zum Teil chaotischen Zustände (mehr als 400 Programmiersprachen allein im Auftragsbereich der US-Rüstungsindustrie) auf dem Softwaresektor durch eine streng standardisierte, modernsten Anwendungen Rechnung tragende Hochsprache zu überwinden. Gegen Ende der 80er Jahre kann eingeschätzt werden, daß das eigentliche Entwicklungsziel, die Bereitstellung programmiersprachlicher Grundlagen für eine moderne Softwaretechnologie, erreicht wurde. Das wird vor allem in der immer stärkeren Hinwendung des Hochtechnologiebereichs zu Ada sichtbar.

Ada muß als Programmiersprache angesehen werden, die sich qualitativ von den bisher verbreiteten höheren Programmiersprachen unterscheidet. Sie enthält neue Sprachkonzepte, die sowohl den gestiegenen Möglichkeiten der Hardware als auch dem wachsenden Komplexitätsgrad der Anwendungen Rechnung tragen. Programmiersicherheit, Maschinenunabhängigkeit und Effektivität sind zentrale Anliegen von Ada, Echtzeit- und Parallelverarbeitung die entscheidenden Anwendungsgebiete. Der „Ada-Stil" ist ein großer Fortschritt auf dem Gebiet der Programmiermethodik.

In der DDR kann für den Beginn der 90er Jahre mit der Verfügbarkeit (und auch dem technologisch bedingten Zwang zur Anwendung) von Ada gerechnet werden. Internationale Erfahrungen zeigen, daß die rechtzeitige Qualifizierung der Ada-Programmierer gegenwärtig zu einem Schlüsselproblem wird.

Hier sollen einige Probleme der Programmierung unter dem Blickwinkel von Ada betrachtet werden, um dem Leser die Einordnung vor dem Hintergrund eigener praktischer Programmiererfahrung zu ermöglichen und sein Interesse für neue Konzepte zu wecken, die bereits diesseits von Echtzeit- und Parallelverarbeitung für den Softwareingenieur von Bedeutung sind. Weitergehende Darstellungen der Sprachkonzepte sind in [4] bis [6] enthalten. In [3] und [11] wird ein in der DDR für 32-bit-Rechner des SKR entwickeltes Ada-O-System vorgestellt, das eine Untermenge von Ada implementiert.

1. Stabile Software

Der Programmierer geht davon aus, daß seine Programme unter gewissen Bedingungen laufen werden. Dazu gehören z. B. eine erwartete Eingabe, die Verfügbarkeit von Ressourcen, aber auch die Korrektheit nachgenutzter Software. Diese Bedingungen werden (wenn überhaupt) als Kommentare und/oder in Nutzerhandbüchern formuliert und sind somit schwer zu finden und leicht zu ignorieren. Die „klassische" Reaktion auf in diesem Sinne unvorhergesehene Ereignisse ist der Absturz des Programms. Ada bezieht von vornherein unvorhergesehene Ereignisse als sogenannte **Ausnahmen** (exceptions) in die Programmierung ein.

Beispiel:

```
(01) with TEXT_IO; use TEXT_IO;
(02) function In_Natural return Natural is
(03)        I: Natural; ——alle Variablen muessen deklariert werden
(03) begin
(04)    loop
(05)       begin
(06)          Get (I);  ——hier koennen Ausnahmen entstehen
(07)          return I;——korrektes Verlassen des Unterprogramms
(08)       exception ——ausnahmebehandlung
(09)       when DATA_ERROR! CONSTRAINT_ERROR=>
                                            ——Eingabe konnte
(10)       ——nicht als natuerliche Zahl interpretiert werden
(11)          Skip_Line; ——und wird ueberlesen
(12)          Put_Line („es wird eine natuerliche Zahl erwartet");
(13)       end;
(14)    end loop;
(15) end In_Natural;
```

Die Funktion **In_Natural** fordert in einer unendlichen Schleife die interaktive Eingabe einer natürlichen Zahl. Wenn die Eingabe nicht der Ada-Syntax für ganze Zahlen entspricht (DATA_ERROR) oder den für natürliche Zahlen zugelassenen Wertebereich überschreitet (CONSTRAINT_ERROR), werden der Anweisungsteil, der für die Ausnahmebehandlung sorgt ((08) bis (13)), abgearbeitet und danach die Schleife erneut durchlaufen. Sobald die Eingabe den Erwartungen entspricht, wird die Funktion durch die **return**-Anweisung verlassen.

Ada ermöglicht neben vordefinierten Standard-Ausnahmen (z. B. CONSTRAINT_ERROR bei Verletzung von Wertebereichen, STORAGE_ERROR bei Speicher-Überlauf, TASKING_ERROR bei falscher Behandlung paralleler Prozesse) dem Programmierer die Einführung eigener Ausnahmen, die die „schwachen Stellen" seines Programms kennzeichnen. Nach diesem Prinzip ist auch das Paket TEXT_IO aufgebaut, das z. B. neben DATA_ERROR auch das unerwartete Erreichen des Endes (END_ERROR) oder das falsche Benutzen (USE_ERROR) eines Files anzeigt. Dieses Paket wurde durch eine Kontext-Klausel (01) an die Funktion angeschlossen.

Ein Paket (so werden in Ada die Module genannt) enthält also neben der Definition von Datenstrukturen und Operationen (Unterprogrammen) auch die Situationen, die bei fehlerhafter Arbeit entstehen können. Damit hat der Programmierer die Möglichkeit, beim Einsatz des Pakets auf solche Situationen zu reagieren. Diese Stabilität ist besonders für Automatisierungs-Software von entscheidender Bedeutung (das Ausbleiben eines Interrupts darf nicht zum Stillstand einer flexiblen Fertigungslinie, Division durch Null nicht zum Verzicht auf die weitere Navigation eines Flugzeugs führen).

Dr. rer. nat. MARTIN HARTWIG (37) studierte von 1970 bis 1975 an der Universität in Rostov am Don/UdSSR Mathematik. Danach von 1975 bis 1983 wissenschaftlicher Mitarbeiter am Institut für Mathematik, seit 1983 am Zentralinstitut für Kybernetik und Informationsprozesse der AdW der DDR. 1983 Promotion A.

Dipl.-Math. ECKHARD STEIN (36) studierte von 1971 bis 1975 Mathematik an der Humboldt-Universität zu Berlin. Von 1975 bis 1983 am Institut für Mathematik und seit 1983 am Zentralinstitut für Kybernetik und Informationsprozesse als wissenschaftlicher Mitarbeiter tätig.

Dr. rer. nat. ROLAND STROBEL (52) studierte von 1955 bis 1960 an der KMU Leipzig Mathematik. Danach von 1960 bis 1962 Tätigkeit im VEB Carl Zeiss JENA. Von 1962 bis 1983 im Institut für Mathematik und seit 1983 im Zentralinstitut für Kybernetik und Informationsprozesse der AdW der DDR als wissenschaftlicher Mitarbeiter tätig. 1969 Promotion A.

[1] Ada ist ein eingetragenes Warenzeichen des U.S. Government Ada Joint Program Office

2. Sicheres Programmieren

Die Programmiersicherheit ist für den gesamten Lebenslauf der Software von Bedeutung. Ein Ada-Anwender (*Lockeed Missiles & Space Company* [9]) rechnet beim Umstieg von seiner bisherigen Systemsprache JOVIAL auf Ada für ein einziges konkretes Anwendungsprojekt (Laufzeit vom Jahr 1992 bis etwa 2010, Programmumfang von anfänglich 500000 bis endlich 1580000 Zeilen) aufgrund bisheriger Erfahrungen mit Einsparungen durch höhere Programmiersicherheit bis zu 74 Millionen US-Dollar. Diese Zahlen sagen natürlich in erster Linie etwas über den Umfang des Projekts aus. Sie machen aber auch deutlich, daß die Frage der Programmiersicherheit nicht durch die Aufforderung an den Assembler- oder PASCAL-Programmierer zur „strukturierten Programmierung" gelöst werden kann. Komplexe Aufgaben verlangen zuverlässiges und leistungsfähiges Arbeitswerkzeug.

Für den FORTRAN-Programmierer gibt es keine Möglichkeit, die Unterscheidung zwischen Meter und Kilometer dem Compiler anzuvertrauen. Er muß mit dem Typ Integer auskommen und fleißig kommentieren. Der Ada-Programmierer kann sich auf eine strenge automatische Typkontrolle auch über die Grenzen von Paketen hinweg verlassen:

```
(20) type Meter is new Integer;
(21) type Kilometer is new Integer;
(22) type Quadratmeter is new Integer;
(23) function Kilometer_in_Meter (km:Kilometer)return Meter is
(24) begin
(25)    return Meter (1_000*km);
(26) end;
(27) function „*" (Left: Meter; Right: Meter)
(28)      return Quadratmeter is
(29) begin
(30)   return Quadratmeter (Integer (Left) * Integer (Right));
(31) end „*";
——.........
(32) m: Meter:= 10;
(33) km: Kilometer:= 1;
(34) m2: Quadratmeter;
——.........
(36) km:= km+m; ——ein Typ-Fehler bei der Uebersetzung
(37) m:= 2*m;      ——Operation „*" mit Ergebnistyp Meter
(38) m2:= m*m;   ——Operation „*" mit Ergebnistyp Quadrat-
                                                          meter
(39) m:= Kilometer_in_Meter (km); ——explizite Konvertierung
```

Die Einführung der abgeleiteten Typen **Meter** und **Kilometer** führt in Ada zur automatischen Übernahme der vordefinierten Operationen (z. B. „=", „+"). Die Definition neuer Operationen („*") sichert eine flexible Programmierung auch bei strenger Typkontrolle. Das Überladen von Unterprogrammen (Identifizierung von „*" in (37) und (38) anhand der Parameter- und Ergebnistypen) sowie ausgefeilte Sichtbarkeitsregeln erhöhen die Änderungsfreundlichkeit beträchtlich. Auch die Möglichkeiten zur Abstraktion von Implementationsdetails müssen in diesem Zusammenhang erwähnt werden:

```
(40) with Elemente; ——definiert Element_Typ
(41) use Elemente; ——nun wird Element_Typ direkt sichtbar
(41) package LIFO_Schlange is
(43)      function Leer return Boolean;
(44)      function Voll return Boolean;
(45)      procedure Anhaengen (E: Element_Typ);
(44)      procedure Entfernen (E: out Element_Typ);
(45)     Schon_Voll, Noch_Leer: exception;
(46) end LIFO_Schlange;
```

Das Paket **LIFO_Schlange** spezifiziert eine Warteschlange, die verwalteten Elemente haben einen Typ, der im Paket **Elemente** vereinbart wurde. Die Implementierung der Warteschlange (ihre Größe, ihr Aufbau als Feld oder vekettete Liste, die LIFO-Reihenfolge) ist „von außen" nicht sichtbar und im Paket-Körper, der getrennt entwickelt und übersetzt wird, versteckt. Der Nutzer dieses Pakets kann ohne Kenntnis dieser Details arbeiten.

```
(47) with LIFO_Schlange;
——...
(49) while not LIFO_Schlange. Voll loop
(50)        LIFO_Schlange. Anhaengen (Neues_Element);
(51) end loop;
```

Die Ausnahmen **Schon_Voll** und **Noch_Leer** schützen ihn vor dem falschen Gebrauch der Operationen **Anhaengen** und **Entfernen**, die Verletzung des LIFO-Prinzips ist ihm nicht mehr möglich. Der Programmierer, der die Zeilen (47) bis (51) schreibt, arbeitet mit wenigstens zwei Paketen: **LIFO_Schlange** und **Elemente**. Woher erfährt er, wovon diese Programme ihrerseits abhängen, wer sagt Bescheid, wenn zur Zeit für eine Programmkomponente keine aktuelle Version existiert oder eine neue Implementierung (z. B. mehr Speicherplatz für Elemente der **LIFO_Schlange** oder Übergang von der Implementierung als Feld zur Implementierung als verkettete Liste) eingebracht wurde? Wer noch nie mit änderungsfreudigen Kollegen an einem größeren Projekt gearbeitet hat oder sich beim Übersetzen oder Verbinden mit veralteten Programmkomponenten herumschlagen mußte, darf diese Fragen ignorieren. Wer aber durch solche Fehler schon viel Zeit bei der Entwicklung und Wartung seiner Programme verloren hat, kann auf die Ada-Programmierumgebung bauen: Ada sieht eine Programmbibliothek vor, die auf die oben gestellten Fragen Antworten geben kann. Die enge Verbindung zwischen Sprache und Programmierumgebung ist eines der neuen Konzepte aus Ada, die die sprachlichen Ausdrucksmittel zur sicheren Programmierung ganz wesentlich erweitern (ein Beispiel für eine leistungsfähige Ada-Programmbibliothek findet man in [3]).

3. Ausdrucksstärke

Wenn man in FORTRAN, PASCAL oder MODULA einfache Listen anlegen will, muß man dem Compiler beim Programmieren die Länge des zu reservierenden Speicherbereichs mitteilen. Wenn man viele Listen füllen muß und jedesmal zur Sicherheit eine möglichst große Länge angibt, wird auch der größte Hauptspeicher schnell erschöpft sein. Daher organisieren viele Programmierer intern eine eigene Listenverwaltung, die eine bessere Speicherauslastung ermöglicht. Ada befreit von diesem zusätzlichen Programmieraufwand: Es ist möglich, die Länge eines Feldes erst zur Laufzeit anzugeben. Hier soll dafür die Funktion **In_Natural** genutzt werden. Diese Funktion soll zunächst mit einem Parameter ausgerüstet werden, der zur interaktiven Eingabe auffordert. Dazu werden (02) und (06) modifiziert.

```
(02.1) function In_Natural (Prompt: String:=„>> ")
(02.2)                                      return Natural is
(06.1)        Put (Prompt); Get (I);
```

Die Vorbesetzung des Eingabeparameters **Prompt** ermöglicht es, beim Aufruf auf die Angabe eines aktuellen Parameters zu verzichten (wer schon einmal Systemrufe programmiert hat, bei denen für dutzende Parameter die Vorbesetzungen ausreichen, wird diesen der Macrotechnik entlehnten Komfort schätzen).

```
(60) L: array (In_Natural („untergrenze")..In_Natural
(61)             („obergrenze")) of Character:= (others=>'x');
```

In (60) und (61) wird während des Programmlaufs ein Feld angelegt, dessen Speicherbedarf anhand der Eingabe bestimmt wird. Diesem Feld wird sofort ein Anfangswert zugewiesen (die Feld-Elemente werden mit „x" initialisiert). Um mit den bei der Übersetzung noch nicht bekannten Feldgrenzen arbeiten zu können, sieht Ada Attribute vor (z. B. 'FIRST, 'LAST, 'RANGE).

Die Arbeit mit dynamischen Feldern ist sorgfältig in die anderen Sprachkonzepte von Ada eingepaßt: ein Verbund (record) kann dynamische Felder als Komponenten enthalten und wird damit selbst zu einer dynamischen Struktur; als Komponenten einer dynamischen Struktur sind z. B. parallele Prozesse (tasks) zugelassen; mit Hilfe von Zeigern können auf der Halde angelegte dynamische

Strukturen zu Geflechten verknüpft werden; Wertebereichsverletzungen werden automatisch als Ausnahmen (exception CONSTRAINT_ERROR) zur Behandlung gemeldet. Schon dieser kleine Einblick macht klar, daß Ada über eine große Ausdrucksstärke verfügt. Ebenfalls weit über das in konventionellen Sprachen Gebotene hinaus reichen die Konzepte für:

- Gleitkomma- und Festpunkt-Arithmetik
- Parallel- und Echtzeitverarbeitung
- maschinennahes Programmieren (z. B. Beschreibung von Steuerblöcken als Ada-Verbunde oder Abbildung von Interrupts auf Task-Eingänge).

Die Vielzahl ausdrucksstarker Konzepte ist eine Hürde für das Erlernen von Ada. Aber ein Programmierer kann sich die Sprache auch auf dem Niveau von PASCAL (besser noch MODULA) erschließen und dann entsprechend seinem Aufgabenkreis tiefer eindringen. Solche Kapitel, wie Parallel- und Echtzeitverarbeitung oder maschinennahes Programmieren, können sogar vollständig ausgeklammert werden. Mindestens zwei Motive sollten jedoch Bedingung sein, wenn man in die Sprache eindringen möchte: professioneller Ehrgeiz und das Gefühl der Unzulänglichkeit bisher verwendeter Programmiersprachen.

4. Abstraktion

Große Programmsysteme sind komplizierte Gebilde. Ihr korrektes Funktionieren ist meist nur eine durch (mehr oder weniger) intensives Testen begründete Hypothese, die Arbeitsweise in einigen Fällen auch für ihre Autoren schon bald ein Geheimnis. Bei der Beherrschung komplexer Sachverhalte ist die menschliche Fähigkeit zur Abstraktion entscheidend. Daher wurde die Entwicklung der Programmiersprachen in den letzten 10 Jahren von der Suche nach sprachlichen Mitteln geprägt, die es ermöglichen, das Programmieren als einen Abstraktionsprozeß zu erfassen.
Beim Entwurf von Ada war dieser Gesichtspunkt für das Einpassen einzelner Konzepte in die Sprache entscheidend. Hier soll kurz angedeutet werden, wie z. B. das Typ-Konzept auf die Möglichkeiten zum abstrakten Programmieren abgestimmt ist. Pakete als Mittel zur Datenabstraktion wurden am Beispiel **LIFO_Schlange** bereits dargestellt. Hier nun eine weitere Abstraktionsstufe:

```
generic
  type Element_tp is private; ——ein Typ als Parameter
  with function „+“ (X, Y: Element_tp) return Element_tp is < >;
                              ——eine Operation als Parameter
package Matrix_Paket is
type Matrix is array (Integer range < >, Integer range < >)
                              of Element_tp;
  Unvertraeglich: exception;
  function „+“ (M1, M2: Matrix) return Matrix;
end Matrix_Paket;
```

Dieses Paket ist parametrisiert durch den Typ **Element_tp** und ein an diesen Typ gebundenes „+“. Es stellt den neuen Typ **Matrix** zusammen mit einem entsprechenden „+“ zur Verfügung. Die „abstrakte“ Implementierung erfolgt im Paketkörper.

```
package body Matrix_Paket is
function „+“ (M1, M2: Matrix) return Matrix is
  M: Matrix (M1' RANGE (1), M1' RANGE (2);
  begin
    if M1' FIRST (1) /= M2' FIRST (1) or
       M1' FIRST (2) /= M2' FIRST (2) or
       M1' LAST (1) /= M2' LAST (1) or
       M1' LAST (2) /= M2' LAST (2) then
         raise Unvertraeglich; ——explizite Ausnahme
  end if;
  for I in M1' RANGE (1) loop    ——Zeilenweise
    for J in M1' RANGE (2) loop ——Spaltenweise
      M (I, J):= M1 (I, J)+ M2 (I, J);
    end loop;
  end loop;
  end „+“;
end Matrix_Paket;
```

Nun können Pakete für die Behandlung von Matrizen mit beliebigem **Element-Typ** generiert werden.

```
package Int_Matrix is new Matrix_Paket (Integer);
——im generierten Paket-Koerper wird Integer—„+“ eingesetzt
package Float_Matrix is new Matrix_Paket (Float);
——im generierten Paket-Koerper wird Float—„+“ eingesetzt
use Int_Matrix; use Float_Matrix;

MI: Int_Matrix. Matrix (0..5, 0..5);
MF: Float_Matrix. Matrix (1..10, 1..20);
—— ...
MI:=MI+MI;       ——Addition von Integer-Matrizen
MF:=MF+MF;       ——Addition von Float-Matrizen
```

Parametrisierte Programmbausteine, z. B. **Matrix_Paket**, werden in Ada als Schablonen bezeichnet. Die Generierung konkreter Pakete (**Int_Matrix** und **Float_Matrix**) nach solchen Schablonen erfolgt während der Übersetzung (in grober Näherung etwa wie beim Expandieren von Macros). Dieses Konzept ist besonders für problemspezifische Spracherweiterungen wichtig (z. B. wird die gesamte Ein- und Ausgabe in Ada durch vordefinierte Schablonen beschrieben). Es ist auch die Voraussetzung für die universelle Wiederverwendbarkeit von Softwarekomponenten. Das Schreiben von Schablonen ist die „hohe Schule“ der Ada-Programmierung, ihre effektive Behandlung eines der vielen schwierigen Probleme beim Bau von Ada-Compilern.

Schlußbemerkungen

Die Arbeit mit Ada fordert (und fördert) eine neue Programmiermethodologie. Damit kann man sich nur durch praktisches Programmieren vertraut machen. Eine erste Möglichkeit dazu bietet das am Zentralinstitut für Kybernetik und Informationsprozesse der AdW der DDR entwickelte Ada-O-System. Es ist auf den 32-bit-Rechnern SM 52/12 und K 1840 des SKR (Host und Target) verfügbar. Schwerpunkt bei diesem System ist die effektive Implementierung einer Teilmenge, die sich an den Möglichkeiten zur sequentiellen Programmierung in MODULA orientiert (u. a. ohne Echtzeit- und Parallelverarbeitung, ohne Programmschablonen) und das Programmieren im Ada-Stil (mit vollständigem dynamischem Typkonzept) ermöglicht [6].
Zur Zeit gibt es etwa 50 validierte Ada-Implementationen, in der Mehrzahl auf 32-bit-Rechnern. Sie sind zum Teil als Cross-Systeme angelegt und unterstützen u. a. die Codegenerierung für 8-bit-Prozessoren. Da das Konzept zur Parallelverarbeitung in Ada von einer „logischen“ Parallelität ausgeht, die vom Laufzeitsystem auf die verfügbaren Prozessoren abgebildet werden muß, wird eine echte Parallelverarbeitung nur für ausgewählte Zielsysteme realisiert (z. B. COLUMBUS-Projekt der *European Space Agency*).
Viele Probleme der Übersetzung und der Programmierunterstützung sind noch nicht befriedigend gelöst. Fortschritte sind hier zum einen von Hardwareentwicklungen (z. B. für Prozeßsynchronisation und Wertebereichsüberwachung), zum anderen von neuen Übersetzungs- und Optimierungstechniken zu erwarten. Trotzdem wurde bereits für 1986 bei Ada-Software mit einem Umsatz von über einer Mrd. US-Dollar gerechnet [10]. Aus einer Reihe von Studien zu Managementproblemen in der Informationstechnologie [2] geht hervor, daß der Übergang zu Ada ein sorgfältig zu planender Schritt ist, der sich (im Unterschied zur Verbreitung solcher Sprachen wie BASIC oder PASCAL) nicht spontan auf Initiative der Programmierer vollzieht. Die bereits umfangreichen praktischen Erfahrungen zeigen: Ada ist kein Wundermittel, dafür aber eine tragfähige sprachliche Grundlage für eine effektive, an den modernen Anforderungen komplexer Systementwicklung orientierte Softwaretechnologie. Die Verbindung von Ausdrucksstärke, Programmiersicherheit und leistungsfähiger Programmierumgebung führt zu einem beachtlichen Zuwachs an Produktivität.

(Fortsetzung auf S. 245)

F. Baldeweg, J. Bechstein, U. Fiedler

Zur Anwendung assoziativer Neuronennetze für die Diagnose und Therapiesteuerung

0. Einleitung

Neuronennetze (neuronale Strukturen) sind Gegenstand intensiver Forschung. Mit ihnen verbinden sich Vorstellungen zur Behandlung einer Klasse informationeller Prozesse, die außerordentlich hohe Anforderungen an die Leistungsfähigkeit (Entscheidungsfähigkeit, Echtzeitfähigkeit) informationeller Technik stellen. Dies sind Aufgaben, die mit gegenwärtig verfügbaren, aber auch mit den avisierten wissensverarbeitenden Systemen nicht oder nur eingeschränkt realisiert werden können. Dazu gehören Prozesse der Mustererkennung im besonderen, der komplexen Entscheidungsfindung in Echtzeitsystemen im allgemeinen. Die Aufgaben schließen die breitbandige Signalaufnahme und -verarbeitung optischer, akustischer und anderer „Reize" sowie das Erkennen von Formen und dynamischen Wahrnehmungsfeldern in Echtzeit ein [4].
Die Forschung zum Arbeitsgegenstand hat folgende Aspekte:

— Analyse biologischer neuronaler Systeme
— ihre mathematische, physikalische, technische Modellierung
— ihre technische Adaption, durch Simulation, Emulation, Realisierung in VLSI-Technik, in optischen Systemen u. a.

Vorliegender Beitrag formuliert eine Teilaufgabe der Prozeßüberwachung und -sicherung (PÜ/PS): die Erkennung und Aktivierung von Mustern im Rahmen der Diagnose und Therapiesteuerung und ihre prinzipielle Lösung mit Hilfe einer spezifischen technischen Adaption neuronaler Strukturen, einem speziellen assoziativen Neuronennetz.

Prof. Dr. sc. nat. Frank Baldeweg (52) studierte von 1954 bis 1960 an der TH Dresden. Danach im Zentralinstitut für Kernforschung (ZfK) Rossendorf der AdW der DDR tätig. 1967 Promotion A, 1970 Promotion B. Seit 1974 nebenamtlicher Hochschullehrer. Von 1983 bis 1985 Direktor F/E des Kombinates Automatisierungsanlagenbau und Direktor des Instituts für Elektro-Anlagen Berlin. Seither wieder im ZfK Rossendorf.

Dipl.-Math. Jürgen Bechstein (38) studierte von 1969 bis 1974 Mathematik an der TU Dresden. Seither wissenschaftlicher Mitarbeiter im Zentralinstitut für Kernforschung (ZfK) Rossendorf.

Dr.-Ing. Uwe Fiedler (29) studierte von 1977 bis 1982 an der Sektion Informationsverarbeitung der TU Dresden. Danach dort bis 1985 Forschungsstudium, Abschluß Promotion A. Seither wissenschaftlicher Mitarbeiter im Zentralinstitut für Kernforschung (ZfK) Rossendorf.

1. Aufgabenstellung

Der Trend in der Entwicklung von Systemen der Prozeßüberwachung und -sicherung (PÜ, PS) komplexer technischer und technologischer Anlagen deutet auf zunehmende Integration echtzeitfähiger intelligenter informationeller Systeme (IS) hin. Das betrifft insbesondere die u. a. in [1] zusammengefaßten und erläuterten Aufgaben eines IS für die PÜ und PS, wie

— die Signalisation (Erkennen)
— die Bewertung und Diagnose von Störungen
— das Ermitteln und Auslösen von Therapiemaßnahmen.

Spezifische Methoden der künstlichen Intelligenz, insbesondere Expertensysteme, werden entwickelt und auch schon vereinzelt genutzt, um für den Anlagenfahrer den mit der Betriebsführung verbundenen Problemlösungsaufwand zu reduzieren, zudem die Kommunikation mit der Anlage bedienfreundlicher zu gestalten [2].
Es gibt dennoch spezielle Aufgaben im Rahmen der Echtzeitdiagnose, z. B. die Erkennung relevanter Muster (im Zeit-, im Bildbereich), die außerordentlich hohe Anforderungen an die Leistungsfähigkeit des IS stellen, die durch gegenwärtig verfügbare Expertensysteme nicht erfüllt werden können. Dazu sind spezielle parallel arbeitende Strukturen erforderlich, die zudem robust sind.
Ansätze zur Lösung bieten die gegenwärtig wieder recht stark in das Blickfeld gerückten „künstlichen Neuronen" oder deren Strukturen („Neuronale Netze") [3] und [4]. Neuronenmodelle gibt es seit Anfang der 40er Jahre; ihre wirksame technische Realisierung scheiterte bislang am physischen Aufwand, ein Problem, das sich mit der Verfügbarkeit der VLSI-Technik aufzuheben beginnt. Die technische Adaption der in biologischen Systemen erkannten informationellen Funktionen bleibt dennoch eine Aufgabe, die sich über viele Jahre hinziehen wird. Dieser Beitrag hat Überlegungen zum Inhalt, die auf eine solche Adaption, z. B. eines assoziativ und parallel arbeitenden Neuronennetzes, für Aufgaben der PÜ und PS, speziell für Diagnose und Therapiesteuerung, orientieren. Er gibt Hinweise für eine Implementierung derartiger Systeme. Nach einer Betrachtung zur Modellierung von Diagnose und Therapiesteuerung werden Möglichkeiten zur Anwendung von Neuronennetzen diskutiert. Es werden Aspekte der Umsetzung der in [5] bis [7] dargelegten methodischen Betrachtungen zum Konzept des künstlichen Neurons für die Aufgaben der Prozeßautomatisierung diskutiert. Das Prinzip des Neurons und dessen mögliche Nutzung werden anhand eines einfachen Beispiels erläutert.

(Fortsetzung von S. 244)

Literatur

[1] The Programming Language Ada Reference Manual. Lecture Notes in Computer Science 155. Berlin/Heidelberg/New York: Springer-Verlag 1983.
[2] Wallis, J. L. (ed.): Ada: Managing the Transition. Proceedings of the Ada-Europe International Conference, May 1986. Cambridge University Press 1986.
[3] Strobel, R.; Hartwig, M.; Stein, E.: Ada-O Nutzerhandbuch. ZKI-INFORMATIONEN, Sonderheft 2/87.
[4] Hartwig, M.; Stein, E.: Ada — Programmiersprache mit neuen Konzepten. edv-aspekte (1982) 3, S. 51—54.
[5] Hartwig, M.; Stein, E.: Parallelverarbeitung in Ada. rechentechnik/datenverarbeitung (1983) 12, S. 5—7.
[6] Hartwig, M.: Dynamische Typen in Ada. rechentechnik/datenverarbeitung (1988) 1, S. 25—27.
[7] Barnes, J.; Fisher, G. (eds.): Ada in Use. Proceedings of the 1985 Ada International Conference. Cambridge University Press 1985.
[8] Gaumer, D. J.: The Magnavox Transition to Ada. In [2], pp. 43—52.
[9] Arkwright, T.: Global Issues in Reuse from a Real Project. In [2], pp. 13—22.
[10] Rogers, M. W.: IT's Companies Acceptance and Attitude Towards Ada. In [7].
[11] Hartwig, M.; Stein, E.; Strobel, R.: Eine Programmiersprache wird vorgestellt: Ada. rechentechnik/datenverarbeitung (1988) 8, S. 5—21.

msr 8838

Es wird die prinzipielle Funktionsweise demonstriert, es soll aber auch sichtbar gemacht werden, daß Echtzeitfähigkeit und Robustheit im gewünschten Sinne beeinflußbar sind. Solcherart realisierte Netze sollten als Teilsysteme einer durch das PÜ/PS-System getragenen Entscheidungshierarchie, insbesondere im prozeßnahen Bereich eingelagert, betrachtet werden.

2. Modellierung des technologischen Systems

Vorliegende Modellierung für den Zustand realer technologischer Systeme (Basissystem, BS) nutzt eine Notation so, wie in [8] eingeführt, d. h., der Zustand des BS wird durch den Zustandsvektor

$$z(t) \triangleq (z_i)_{i=1,n} \in XZ^n \equiv \boldsymbol{Z}, \qquad i = 1(1)n$$

n Dimension des Zustandsvektors

beschrieben.

Ein Zustand z werde durch die Analyse eines aktuellen Ereignismusters (z. B. binäres Muster) erkannt und bewertet, d. h.

$$z(t) \in \boldsymbol{Z} \subseteq \boldsymbol{B} \times \boldsymbol{N} \times \boldsymbol{T}.$$

$\boldsymbol{T}$: Menge der Zeitwerte

$\boldsymbol{N}$: Bewertungsmenge mit $N \in \{0, 1\}$

$\boldsymbol{B}$: Bezeichnungsmenge

Sowohl Prozeßdaten (dynamische Datenbasis) als auch das Prozeß-/Systemwissen (Modell-/Wissensbasis) sollen zunächst auf das binäre Alphabet abgebildet sein.

Die Bewertung eines Zustands entspräche damit der Abbildung von

$$z(t) \to \boldsymbol{B}_{kl} \equiv \{\text{normal, gestört}\}, \qquad \text{z. B.} \equiv \{0, 1\}.$$

Diese Abbildung repräsentiert die Bewertung des Anlagenzustandes durch gespeichertes (möglicherweise erlerntes) Wissen. Dabei werde ein aktuelles Meßwertmuster mit einer Struktur von Mustern, die die Klasse der anomalen Zustände repräsentiert, verglichen. Diese Bewertung kann die Zuordnung entsprechender Therapieaktionen einschließen.

3. Das künstliche Neuron

Als eine mögliche konkrete Realisierung für ein künstliches Neuron ist aus [9] und [10] das Schwellwertelement bekannt. Um nachfolgende Betrachtungen plausibel zu machen, wird ein solches Schwellwertelement kurz erläutert. Der Schwellwertbaustein (Bild 1) hat n Eingänge und einen Ausgang. Die Eingangswerte e_i, z. B. $e_i \in \{0, 1\}$, $i = 1, \dots, n$, bilden den Eingangsvektor $\boldsymbol{E} = (e_1, e_2, \dots, e_n)$. Die Gewichte g_i, z. B. $g_i \in \{-1, 0, 1\}$ stellen den Gewichtsvektor $\boldsymbol{G} = (g_1, g_2, \dots, g_n)$ dar.

Das Skalarprodukt

$$s = \boldsymbol{E} \cdot \boldsymbol{G}^T = \sum_{i=1}^{n} g_i e_i$$

wird mit dem Schwellwert q verglichen und für den Ausgangswert a soll gelten:

$$a = \begin{cases} 0, & \text{falls } s < q \\ 1, & \text{falls } s \geqq q. \end{cases}$$

Der Schwellwert q (q sei ohne Beschränkung der Allgemeinheit eine ganze Zahl) und die Gewichte g_i können bei jedem Baustein gewählt werden. Mit dem beschriebenen Baustein kann ein beliebiges duales Wort der Länge n erkannt werden.

Beispiel: n = 8.

Der Baustein soll alle Eingangsvektoren der Art (0, *, 0, 1, *, *, 0, 1) erkennen und in diesem Fall den Wert $a = 1$ ausgeben. * bedeutet dabei, daß der spezielle Eingabewert beliebig 0 oder 1 sein kann. Die Lösung ist im Bild 2 dargestellt.

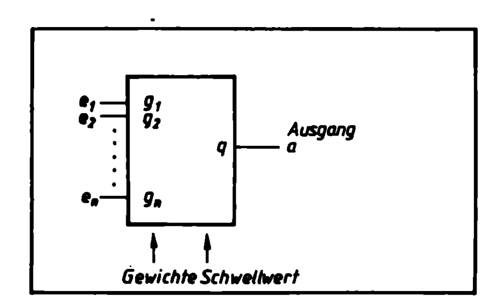


Bild 1. Schwellwertbaustein

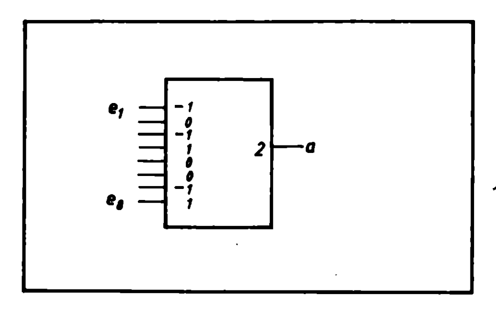


Bild 2. Beispiel eines belegten Schwellwertbausteins

Im folgenden werden bei den Zusammenschaltungen von solchen künstlichen Neuronen alle Eingänge, die mit dem Gewicht 0 versehen sind, nicht mehr mitgezeichnet, da sie das Resultat nicht beeinflussen.

4. Anwendung von Neuronennetzen

4.1. Zustandserkennung und -bewertung

Die Bewertung von Zuständen eines technologischen Systems erfolgt unter Zuhilfenahme von speziellem, in geeigneter Weise fixiertem Wissen. Dafür kommen Neuronennetze durchaus in Frage.

Unter einem Neuronennetz soll dabei im besonderen eine Struktur über einer Menge von Schwellwertbausteinen verstanden werden. Damit kann der Zusammenhang zu den in [11] und [12] definierten

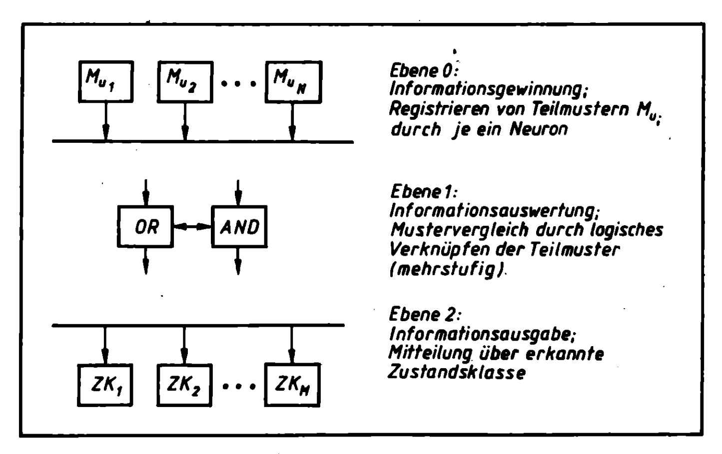
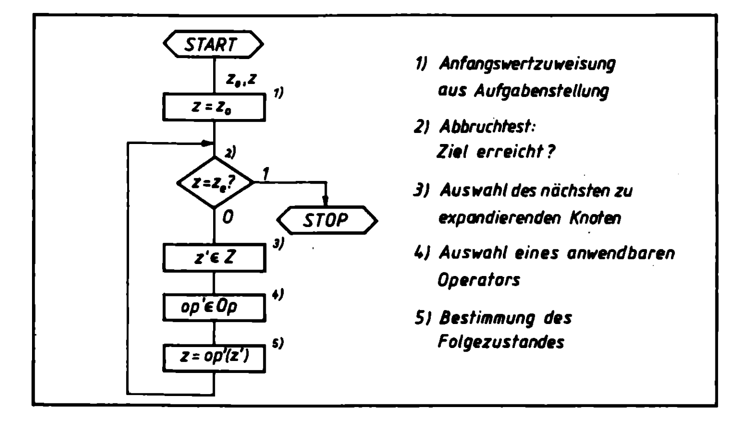


Bild 3. 3-Ebenenstruktur der Zustandserkennung

Bild 4. Suche im Zustandsgraphen



Begriffsnetzen hergestellt werden. Es ist zudem vorstellbar, daß ein Neuron auf einer unteren Hierarchieebene sogenannte informationelle Invarianten repräsentiert. Da jegliche Situation mehrschichtig bewertet werden kann, ist es möglich, daß Neuronennetze in einer unteren Schicht zu Invarianten einer höheren Schicht zusammenfaßbar sind.

Folgendes sei zu beachten:

- Der Zustand des technologischen Systems werde vereinfachend durch ein endliches binäres Muster $z(t)$ repräsentiert.
- Neuronen klassifizieren und erkennen dieses binäre Zustandsmuster.
- In realen technischen Systemen ist die Dimension von $z(t) \in Z$ recht groß.
- Neuronen können auch geringfügig verfälschte Eingabemuster einordnen.
- Die Menge Z kann zerlegt werden:

$$Z = \bigcup_i Z_{ki} \quad \text{mit} \quad \bigcap_i Z_{ki} \neq 0 .$$

- Für die Menge aller Muster MU läßt sich schreiben:

$$MU = MU_1 \cup MU_2 \cup \dots \cup MU_n ,$$

wobei jedem MU_i ein Neuron zugeordnet werden kann, das dieses Muster assoziativ zu erkennen in der Lage ist.

Das Problem der Zustandserkennung besteht darin, aktuelle Muster als Teilmuster zu registrieren und zu bewerten, d. h. in die Klassen {normal, anomal} einzuordnen und zu signalisieren, wenn gefährliche Zustände erkannt werden. Eine Lösung für dieses Problem ist im Bild 3 dargestellt. Es handelt sich um ein Mehrstufenkonzept. Es ist vorstellbar, daß in der Menge möglicher Zustände die Teilmenge der gefährlichen Zustände erkannt wird. Das reduziert den Aufwand zur Klassifikation und Signalisation (Ausnahmeprinzip).

4.2. Therapiesteuerung

Therapiesteuerung führt das Basissystem aus einem gestörten Zustand $z(t)$ innerhalb eines Zeitintervalls Δ mit Hilfe einer Strategie U in einen normalen Zustand z_0 [8]. Die Strategie entspräche z. B. einer zeitlich geordneten Folge von Maßnahmevektoren $u(t), \dots, u(t + \Delta)$. Für den konkreten Fall ist die Zuordnung von gestörtem Zustand und Maßnahmevektor (bzw. Strategie) erforderlich. Diese Zuordnung ist technologisch bedingt. Sie kann durch assoziative Neuronennetze vorgenommen werden. Dazu werden $z(t)$ identifiziert und $u(t)$ über das Neuronennetz ausgelöst. Dieses Prinzip, es sei als Therapiesteuerung durch Zuordnung bezeichnet, wird im folgenden Abschnitt anhand eines einfachen Beispiels erläutert. Als ein weiteres Beispiel kann die heuristische Ermittlung von Therapiestrategien genannt werden. Neuronennetze können in diesem Fall dazu dienen, die Mustererkennung im Zustandsraum zu beschleunigen. Der durch die Menge aller möglichen Zustände gebildete Zustandsraum kann dabei formal durch einen Zustandsgraphen dargestellt werden:

$$G_z = [Op, k] .$$

Op Menge von Operatoren; k Menge von Knoten

Die Aufgabe besteht darin, eine technologisch mögliche Folge von Operatoren zu erkunden, mit der ein gegebener Initialzustand z_0 in einen bekannten Zielzustand z_e überführt werden kann (Bild 4). Diese Aufgabe entspricht der Lösung eines Pfadproblems [13]. Randbedingungen sind:

- Der Zustandsraum hat eine große Dimension.
- Es wird ein optimaler Weg gesucht.
- Der Zielzustand wird implizit durch Restriktionen beschrieben (Zustandsklasse).
- Um einen optimalen Weg zu erreichen, ist es nötig, diese Prozeduren mehrmals mit variierenden Auswahlkriterien auszuführen.
- Für die Suche in Zustandsräumen großer Dimension sind heuristische Entscheidungskriterien zu verwenden [11].

5. Beispiel

Mit folgendem Beispiel, das auf jenes in [6] Bezug nimmt, soll das Prinzip der assoziativen Therapiesteuerung demonstriert werden. Die hier verwendete Dimension von $z(t)$ ist nicht signifikant, d. h. enthält eigentlich kein Motiv für die Nutzung des Prinzips. Reale Systeme sind jedoch durch eine weitaus größere Dimension des Zustandsvektors charakterisiert, was die Anwendung von Neuronennetzen sinnvoll erscheinen läßt.

Gegeben sei der Zustand $z(t) \equiv (z_1, z_2, \dots, z_i, \dots, z_n)$, z. B. $z_i \in \{0, 1\}$, des zu steuernden Systems. Ein Teilzustand der Anlage ist eine Teilmenge von Komponenten des Zustandsvektors. Im allgemeinen soll das Neuron einen Teilzustand des Prozesses erkennen und die notwendige Steuerung einleiten, d. h., die entsprechenden Komponenten des Steuervektors $u(t) \equiv (u_1, \dots, u_m)$, $u_i \in \{0, 1\}$, realisieren ($u_i \equiv 1 \triangleq$ „Steuerung ein", $u_i \equiv 0 \triangleq$ „Steuerung aus"). Es ist durchaus möglich, daß mehrere Steuerbefehle auf das gleiche Steuerobjekt wirken, z. B.

$u_1 = 1$, Heizung ein,
$u_2 = 1$, Heizung aus.

Die Zustandsmenge Z sei die Menge aller möglichen Belegungen des Zustandsvektors.

Satz:
Jede beliebige Teilmenge $Z' \subset Z$ der Zustandsmenge Z kann durch eine Neuronenschaltung eindeutig erkannt werden.

Jeder einzelne Zustand $z(t) \in Z'$ läßt sich durch ein spezielles Neuron erkennen. Alle Zustände $z(t) \in Z'$ können dann über ein oder mehrere Neuronen, die als ODER-Struktur arbeiten, zusammengefaßt werden. In der Praxis genügen deshalb i. allg. zwei Schichten von Neuronen, um eine Teilmenge von Z zu erkennen.

Das Prinzip der Steuerung ist Bild 5 zu entnehmen.

Wegen des obigen Satzes kann also mit jeder beliebigen Teilmenge der Zustandsmenge Z jeder beliebige Steuerbefehl $u(t)$ ausgelöst werden.

Bei einander entgegengesetzten Steuerbefehlen ist darauf zu achten, daß die auslösenden Teilmengen disjunkt sind (z. B. „ein"- und „aus"-schalten).

Beispiel: Chemischer Reaktor

Die konkrete Realisierung eines Neuronennetzes zur Bewertung eines Zustandsvektors und Anzeige einer Therapiesteuerung nimmt Bezug auf ein in [6] erläutertes Beispiel, einen hypothetischen chemischen Reaktor.

Der chemische Reaktor, charakterisiert durch den Zustandsvektor $z(t) \in Z$ (Tafel 1) werde durch das IS so gesteuert (Steuervektor $u(t) \in U$, Tafel 1), daß $z \in Z$, d. h. Element der Menge der Normalzustände Z, bleibt.

In Tafel 2 sind die Sensorzustände bezüglich der Temperatur zusammengefaßt.

Die gekennzeichneten Zustände (*) entsprechen vernünftigen, echten Situationen des Systems. Falls diese signalisiert werden, wird gegebenenfalls eine entsprechende Reaktion veranlaßt. Falsche Störmeldungen können z. B. infolge Meßstellenausfalls ausgelöst werden. Sie machen sich als widersprüchliche Sensorzustände bemerkbar.

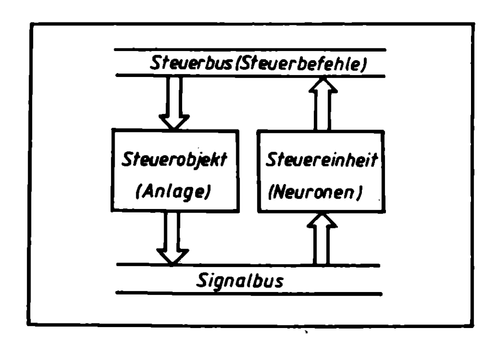
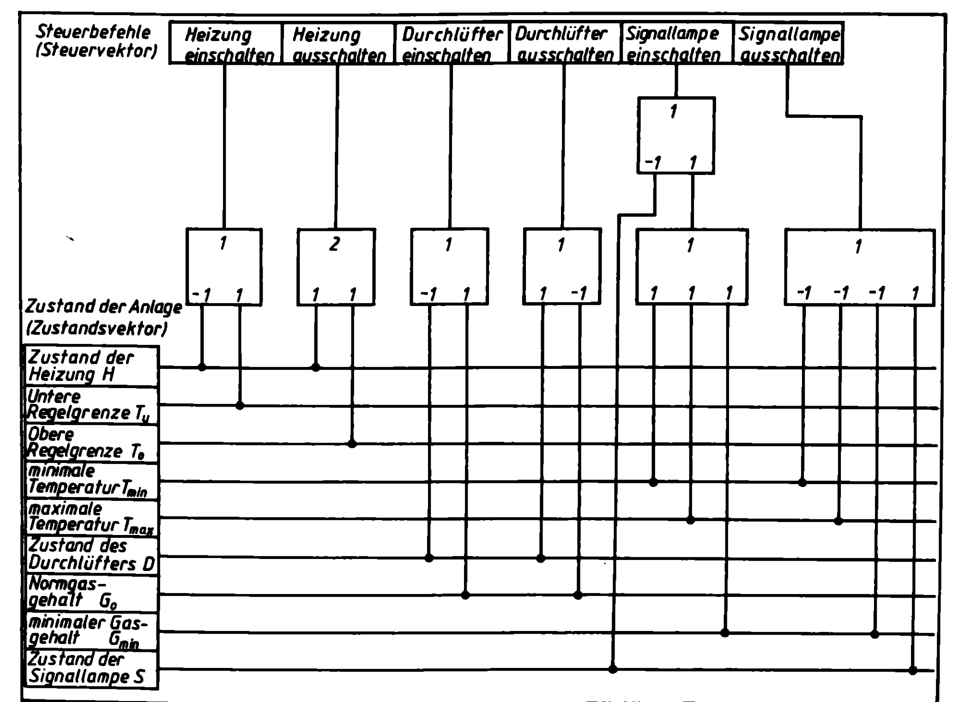


Bild 5. Prinzip der Therapiesteuerung



Tafel 2. Sensorteilzustände bezüglich Temperatur

T_0	T_U	T_{max}	T_{min}	Bewertung
0	0	0	0	Normalzustand (*)
0	0	0	1	WS
0	0	1	0	WS
0	0	1	1	WS
0	1	0	0	Einschalten der Heizung, falls noch nicht eingeschaltet (*)
0	1	0	1	Alarm (*)
0	1	1	1	WS
1	0	0	0	Heizung ausschalten, falls noch nicht ausgeschaltet (*)
1	0	0	1	WS
1	0	1	0	Alarm (*)
1	0	1	1	WS
1	1	0	0	WS
1	1	0	1	WS
1	1	1	0	WS
1	1	1	1	WS

WS Widerspruch in der Sensoranzeige

Bild 6.
Neuronennetz zur Therapiesteuerung eines chemischen Reaktors

Tafel 1. Komponenten des Zustands- und des Steuervektors

Komponenten des Zustandsvektors z(t)

Komponente	Symbol	0	1
1 Heizung	H	aus	ein
2 untere Regelgrenze der Heizung	T_U	$T \geqq T_U$	$T < T_U$
3 obere Regelgrenze	T_0	$T \leqq T_0$	$T > T_0$
4 minimale Temperatur	T_{min}	$T \geqq T_{min}$	$T < T_{min}$
5 maximale Temperatur	T_{max}	$T \leqq T_{max}$	$T > T_{max}$
6 Durchlüfter	D	aus	ein
7 Normgasgehalt	G_0	$G \geqq G_0$	$G < G_0$
8 minimaler Gasgehalt	G_{min}	$G \geqq G_{min}$	$G < G_{min}$
9 Signallampe	S	aus	ein

Komponenten des Steuervektors u(t)

0	1
1 kein Befehl	Heizung einschalten
2 kein Befehl	Heizung ausschalten
3 kein Befehl	Durchlüfter einschalten
4 kein Befehl	Durchlüfter ausschalten
5 kein Befehl	Signallampe einschalten
6 kein Befehl	Signallampe ausschalten

Im gegebenen Fall lassen sie sich durch 6 künstliche Neuronen erkennen:

$$(1, 1, *, *)\,, \quad (*, *, 1, 1)\,, \quad (0, *, 1, *)\,,$$

$$(*, 0, *, 1)\,, \quad (1, *, *, 1)\,, \quad (*, 1, 1, *)\,.$$

Das vereinfacht dargestellte Prinzip der Lösung für das Diagnose- und Therapieproblem zeigt Bild 6.

Zusammenfassung und Ausblick

Die praktische Realisierung des an einem einfachen Beispiel erläuterten Konzepts für das Gebiet der PÜ/PS ist u. a. davon abhängig, ob geeignete Schaltkreise zur Verfügung stehen. Die Dimension der zu verarbeitenden Zustands- und Steuervektoren ist in der Praxis beträchtlich. Dies setzt eine entsprechend hohe Integration der speziellen Bauelemente voraus. Als mögliche Lösung bieten sich semikundenspezifische Schaltungen, z. B. auf der Basis von „gate arrays", an. Dabei würden als Grundelemente die erläuterten Schwellwertbausteine fungieren. Der entscheidende Vorteil solcher Schaltungen gegenüber ähnlichen Lösungen mit Hilfe von Digitalrechnern liegt in der um Größenordnungen höheren Verarbeitungsgeschwindigkeit. Eine höhere Zuverlässigkeit ergibt sich daraus, daß innerhalb eines einzelnen Neurons eine sehr einfache Informationsverarbeitung erfolgt und zudem der Ausfall eines Neurons nicht den Ausfall des Gesamtsystems zur Folge hat.

Die Autoren sind davon überzeugt, daß mit der praktischen Verwirklichung des Konzepts eine wesentliche Verbesserung der Leistungsfähigkeit und Zuverlässigkeit für Systeme der Prozeßüberwachung und -sicherung erreicht wird. Bausteine dieser Art sind u. a. als Bestandteil mehrschichtig geschalteter, entscheidungsunterstützender Beratungssysteme aufzufassen [14].

Literatur

[1] Baldeweg, F.; Balzer, D.; Brack, G.: Automatische Prozeßsicherung in Produktionssystemen. Band 204 der REIHE AUTOMATISIERUNGSTECHNIK. Berlin: VEB Verlag Technik 1984.

[2] Baldeweg, F.; Fiedler, U.; Werner, M.; Weiss, F. P.: Knowledge based Diagnostics in Nuclear Power Plants. In: Proceedings of the ANS „Topical Meeting on Artificial Intelligence in the Nuclear Industry". Snowbird, Utah/USA 31. August bis 2. September 1987.

[3] Lippmann, R.: An Introduction in Computing with Neural Nets. IEEE ASSP Magazine 4 (1987) 2, S. 4—22.

[4] Schulz, W.: In Computern laufen nervliche Vorgänge ab. VDI-Nachrichten Nr. 30/24. Juli 1987, S. 12.

[5] Bechstein, J.: Grundprinzipien der assoziativen Speicherung im menschlichen Gehirn und ihre Anwendung in technischen Systemen. Dissertation A, AdW der DDR 1987.

[6] Bechstein, J.: Ein Neuronenmodell zur Langzeitspeicherung. Vortrag auf dem Kongreß der Gesellschaft für Biomedizinische Technik der DDR, Leipzig 1977.

[7] Bechstein, J.: A Special Neuron Model for the Simulation of the Long-Time Memory of the Human Brain. 5th Conference and Exhibition on Bioengineering, Budapest 1981.

[8] Baldeweg, F.; Gassmann, F. H.: Formale Beschreibung von Diagnose und Therapiesteuerung in einem gestörten diskontinuierlich-diskreten Basissystem. msr, Berlin 23 (1980) 10, S. 575—579.

[9] Langheld, H.: Schwellwertlogik im Nervensystem. Elektronik (1976) 3, S. 99—104.

(Fortsetzung auf S. 249)

F. Beichelt, L. Sproß

Ein Verfahren zur approximativen Berechnung der Zuverlässigkeit komplizierter Systeme

0. Einleitung

Fundierte Aussagen über Zuverlässigkeitskenngrößen technischer Erzeugnisse werden heute von jedem Hersteller und Betreiber gefordert. Derartige Aussagen sind vor allem für komplexe Anlagen, wie vollautomatische Fertigungslinien, Transport- und Sicherungssysteme sowie Kraftwerke, unerläßlich. Die damit verbundenen Probleme sind jedoch nicht nur technisch-physikalischer Natur. Auch rechentechnisch zählt die exakte Berechnung von Zuverlässigkeitskenngrößen eines komplizierten Systems mit zu den schwierigsten Problemen. So ist es gegenwärtig auch beim Einsatz modernster Rechentechnik nicht möglich, Zuverlässigkeitskenngrößen eines Systems aus entsprechenden Zuverlässigkeitskenngrößen seiner Elemente (Teilsysteme) in vertretbarer Zeit exakt zu berechnen, wenn deren Anzahl größer als 30 ist und zuverlässkgkeitstheoretisch eine hinreichend komplizierte „Vermaschung" der Elemente vorliegt [6]. Die Ursache dafür liegt in der mit wachsender Anzahl der Elemente exponentiell zunehmenden Rechenzeit. Dieser Sachverhalt gilt schon für die binären Systeme und damit erst recht für die mehrwertigen (bez. der verwendeten Fachterminologie und den im folgenden verwendeten theoretischen Grundlagen s. [2]). Andererseits sind bei komplexen technischen Systemen auch nach erfolgter modularer Zerlegung, insbesondere nach Parallel- und Serienreduktionen der zugehörigen Zuverlässigkeitsschaltbilder, i. allg. noch weit mehr als 30 Teilsysteme zu berücksichtigen. So enthalten zum Beispiel größere rechnergestützte Steuerungs- und Regelungsanlagen zehn- und hunderttausende elektronische Bauelemente, nach einer modularen Zerlegung sind immer noch hunderte Teilsysteme explizit in die Zuverlässigkeitsanalyse der Anlagen einzubeziehen. Aus diesen Gründen haben solche Verfahren eine besondere Bedeutung erlangt, die zwar nur untere oder (und) obere Schranken für Zuverlässigkeitskenngrößen des Systems liefern, dafür aber nach vertretbaren Rechenzeiten Aussagen für beliebig komplexe Systeme gestatten. In diesem Beitrag wird ein derartiges Verfahren für binäre monotone Systeme vorgestellt, das von minimalen Pfad- und Schnittmengen des Systems ausgeht und das Abschätzungen der Systemverfügbarkeit bei bekannten Verfügbarkeiten der Elemente liefert. Theoretische Überlegungen und die Analyse zahlreicher numerischer Beispiele haben ergeben, daß das Verfahren für eine große Klasse von Systemen hinreichend genaue Abschätzungen der Systemverfügbarkeit in vertretbaren Rechenzeiten liefert, wenn eine exakte Berechnung schon längst nicht mehr möglich ist.

Prof. Dr. sc. techn. Dr. rer. nat. FRANK BEICHELT (46) studierte von 1961 bis 1966 Mathematik an der Friedrich-Schiller-Universität Jena. Seine Arbeitsstätten waren von 1966 bis 1969 die Sektion Forstwirtschaft der TU Dresden, von 1969 bis 1974 die Sektion Mathematik der Bergakademie Freiberg, von 1974 bis 1979 das Braunkohlenkombinat „Gustav Sobottka" in Röblingen/See, von 1979 bis 1981 die Sektion Technische Verkehrskybernetik der HfV „Friedrich List" Dresden und seit 1981 die Sektion Informationselektronik der Ingenieurhochschule Mittweida, wo er als ordentlicher Professor für Mathematik den WB Mathematik leitet. 1972 Promotion A, 1978 Promotion B.

Dr. sc. techn. LUTZ SPROSS (29) studierte von 1980 bis 1985 an der Ingenieurhochschule Mittweida, wo er eine vertiefte Ausbildung im Fach Mathematik erhielt. Von 1985 bis 1988 Forschungsstudent im Wissenschaftsbereich Mathematik an der gleichen Hochschule. 1988 Promotion B (Rechnergestützte Zuverlässigkeitsanalyse großer Systeme). Seit 1988 im VEB ELFEMA Mittweida als Forschungsingenieur tätig.

1. Grundlagen

Das binäre monotone System S bestehe aus den Elementen $e_1, e_2, \ldots, e_n$. Die zufälligen Booleschen Indikatorvariablen für den Zustand des Systems und der Elemente seien X_s bzw. X_i, $i = 1, 2, \ldots, n$:

$$X_s = \begin{cases} 1, \text{ wenn } S \text{ funktionstüchtig}, \\ 0, \text{ wenn } S \text{ funktionsuntüchtig}; \end{cases}$$

$$X_i = \begin{cases} 1, \text{ wenn } e_i \text{ funktionstüchtig}, \\ 0, \text{ wenn } e_i \text{ funktionsuntüchtig}. \end{cases}$$

Jedem Zustandsvektor $X = (X_1, X_2, \ldots, X_n)$ wird durch die Strukturfunktion φ des Systems eindeutig ein Systemzustand zugeordnet:

$$X_s = \varphi(X).$$

Die Verfügbarkeit der Elemente seien $p_i = P(X_i = 1) = E(X_i)$, $i = 1, 2, \ldots, n$; die des Systems sei $p_s = P(X_s = 1) = P(\varphi(X) = 1) = E(\varphi(X))$. Die Kenntnis von φ ist also für die exakte Berechnung der Systemverfügbarkeit erforderlich. Ihre Bestimmung erfolgt i. allg. über die minimalen Pfadmengen $W_1, W_2, \ldots, W_w$ bzw. minimalen Schnittmengen $S_1, S_2, \ldots, S_s$ des Systems. Werden wie üblich mit $X \vee Y = X + Y - XY$ die Disjunktion und mit $X \wedge Y = XY$ die Konjunktion zweier Boolescher (0, 1)-Variabler X und Y bezeichnet und wird

$$A_j = \bigwedge_{i \in W_j} X_i \quad \text{und} \quad B_j = \bigvee_{i \in S_j} X_i \tag{1}$$

gesetzt, so gelten die logisch äquivalenten Darstellungen der Strukturfunktion

$$\varphi = \bigvee_{j=1}^{w} A_j \tag{2}$$

und

$$\varphi = \bigwedge_{j=1}^{s} B_j. \tag{3}$$

(Fortsetzung von S. 248)

[10] KOPPERSCHMIDT, G.: Majoritätslogik in der Schaltungspraxis. Elektronik (1976) 10, S. 71–78.

[11] LISS, E.: Lernender Homöostat zur Demonstration adaptiver Steuerungen mit künstlichem Intellekt. msr, Berlin 25 (1982) 9, S. 489–493.

[12] LISS, E.: Logische Systeme zum strukturierten Speichern und lerntypischen Verwerten. msr, Berlin 25 (1982) 12, S. 669–673.

[13] NILSON, N. J.: Principles of Artificial Intelligence. Berlin/Heidelberg/New York: Springer-Verlag 1982.

[14] FIEDLER, U.; LINDNER, A.; BALDEWEG, F.; KLEBAU, J.: An artificial Intelligence Approach towards Disturbance Analysis. Kernenergie 29 (1986), S. 302.

msr 8837

Durch (2) bzw. (3) ist eine disjunktive bzw. konjunktive Normalform der Strukturfunktion gegeben. Durch Negation von (3) erhält man eine weitere Darstellung für φ:

$$\varphi = 1 - \bigvee_{j=1}^{s} \bigwedge_{i \in B_j} \bar{X}_i , \tag{4}$$

wobei hier und im folgenden die Negation einer beliebigen Booleschen Variablen X mit $\bar{X}$ bezeichnet wird ($\bar{X} = 1 - X$). Anstelle von (2) und (3) hat man auch die äquivalenten Darstellungen

$$\varphi = \max_{1 \leq j \leq w} \min_{i \in W_j} X_i , \tag{5}$$

$$\varphi = \min_{1 \leq j \leq s} \max_{i \in S_j} X_i . \tag{6}$$

Leistungsfähige Verfahren zur Berechnung der Pfad- und Schnittmengen auf der Basis von Zuverlässigkeitsschaltbildern sind in [2] und [7] beschrieben und im Programmpaket [5] für Personalcomputer implementiert (PASCAL- und FORTRAN-Programme).
Die Darstellungen (2) bis (6) der Strukturfunktion erlauben jedoch auch bei Unabhängigkeit der $X_1, X_2, \ldots, X_n$ keine unmittelbare Berechnung der Systemverfügbarkeit. Dazu ist die Strukturfunktion in eine Orthogonalform zu überführen, das heißt in einer zu (2) bis (6) logisch äquivalenten Form

$$\varphi = \sum_{i=1}^{d} D_i \quad \text{mit} \quad D_i D_j = 0 \quad \text{für} \quad i \neq j \tag{7}$$

darzustellen, wobei die D_i Konjunktionen gewisser X_j und $\bar{X}_k$, $j \neq k$, sind. Liegt φ in der Form (7) vor, gilt

$$P_s = E(\varphi) = \sum_{i=1}^{d} E(D_i) . \tag{8}$$

Im Fall der Unabhängigkeit der X_i erhält man also aus einer Orthogonalform die Systemverfügbarkeit dadurch, daß in φ alle X_j durch p_j und alle $\bar{X}_k$ durch $\bar{p}_k = 1 - p_k$ ersetzt werden. Zur Erzeugung von Orthogonalformen sind sogenannte Orthogonalisierungsverfahren entwickelt und rechentechnisch umgesetzt worden. Für die weiteren Ausführungen sind diejenigen Orthogonalisierungsverfahren von Bedeutung, die von der Darstellung der Strukturfunktion in der Form

$$\varphi = \sum_{i=1}^{w} G_i \quad \text{mit} \quad G_i = \bar{A}_1 \bar{A}_2 \ldots \bar{A}_{i-1} A_i \tag{9}$$

ausgehen. (Offenbar sind die G_i paarweise orthogonal, das heißt, es gilt $G_j G_k = 0$ für $j \neq k$.) Zu den numerisch effektivsten Verfahren dieser Art gehören die in [1] [3] [7] bzw. [5] beschriebenen bzw. programmierten.

2. Schranken für die Systemverfügbarkeit

Zu Vergleichszwecken sollen zunächst einige bekannte Schranken für die Verfügbarkeit eines Systems mit unabhängigen Elementen angegeben werden, die auf den minimalen Pfad- und Schnittmengen beruhen.
Aus (5) und (6) folgt für beliebige j und k mit $1 \leqq j \leqq w$ und $1 \leqq k \leqq s$

$$\prod_{i \in W_j} p_i \leqq p_s \leqq 1 - \prod_{i \in S_k} \bar{p}_i . \tag{10}$$

Sind alle p_i einander gleich, so liefern jeweils die Pfad- bzw. Schnittmengen mit der kleinsten Mächtigkeit die besten unteren bzw. oberen Schranken für φ. Allgemein folgt aus (10)

$$\max_{1 \leq j \leq w} \prod_{i \in W_j} p_i \leqq p_s \leqq 1 - \max_{1 \leq k \leq s} \prod_{i \in S_k} p_i \tag{11}$$

Ferner gilt [2]

$$l(p) \leqq p_s \leqq r(p) \tag{12}$$

Tafel 1.
Anzahl minimaler Pfade und Schnitte (Bild 1)

m	w	s
4	3	4
7	21	18
10	151	56
13	1081	148
16	7739	356
19	55405	806

mit $p = (p_1, p_2, \ldots, p_n)$ und

$$l(p) = \prod_{k=1}^{s} \left[1 - \prod_{i \in S_k} \bar{p}_i\right], \qquad r(p) = 1 - \prod_{j=1}^{w} \left(1 - \prod_{i \in W_j} p_i\right).$$

Die untere Schranke $l(p)$ bzw. die obere Schranke $r(p)$ für p_s erhält man formal dadurch, daß in (3) bzw. in (2) die X_i durch p_i und die X_j durch p_j ersetzt werden.
Die Berechnung der Schranken (11) und (12) erfordert die Kenntnis aller minimalen Pfad- und Schnittmengen des Systems. Jedoch wachsen auch deren Anzahlen exponentiell mit der Komplexität des Systems (s. Tafel 1). Daher sind der Anwendung der Schranken (11) und (12) ebenfalls rechentechnische Grenzen gesetzt. Erstrebenswert ist somit die Konstruktion guter Schranken für p_s auf der Basis möglichst weniger minimaler Pfad- bzw. Schnittmengen. Diesem Ziel dienen die folgenden Ausführungen. Ohne Beschränkung der Allgemeinheit wird vorausgesetzt, daß $W_1, W_2, \ldots, W_u$ bzw. $S_1, S_2, \ldots, S_v$ diejenigen minimalen Pfad- bzw. Schnittmengen sind, mit denen die Schranken konstruiert werden sollen; $u \leqq w$, $v \leqq s$. Aus (5) und (6) folgt

$$\max_{1 \leq j \leq u} \min_{i \in W_j} X_i \leqq \varphi \leqq \min_{1 \leq j \leq v} \max_{i \in S_j} X_i .$$

Analog zu (2) bzw. (4) kann man diese Beziehungen in der Form

$$\bigvee_{j=1}^{u} A_j \leqq \varphi \leqq 1 - \bigvee_{j=1}^{v} C_j \tag{13}$$

mit

$$C_j = \bigwedge_{i \in S_j} \bar{X}_i$$

schreiben. Wird

$$H_i = \bar{C}_1 \bar{C}_2 \ldots \bar{C}_{i-1} C_i$$

gesetzt, so erhält man entsprechend (9) die gewünschten Schranken für die Systemverfügbarkeit:

$$\sum_{i=1}^{u} E(G_i) \leqq p_s \leqq 1 - \sum_{i=1}^{v} E(H_i) . \tag{14}$$

Die Berechnung der $E(G_i)$ und $E(H_i)$ erfolgt am zweckmäßigsten durch Konstruktion von Orthogonalformen für G_i und H_i. Hierfür bietet sich der in [3] beschriebene Algorithmus an, der genau auf die vorliegende Situation zugeschnitten ist. Beim Vergleich mit den Schranken (11) und (12) wird deutlich, daß (14) zwar nicht alle minimalen Pfad- und Schnittmengen des Systems erfordert, dafür aber die Konstruktion von Orthogonalformen notwendig macht. Andererseits liefern diese Orthogonalformen zusätzliche Informationen über die Systemverfügbarkeit. Dadurch können die Schranken (14) auch bei relativ kleinen u und v schärfer sein als (11) und (12). Wegen des exponentiellen Wachstums der Anzahl der minimalen Pfade und Schnitte bei wachsender Komplexität des Systems sind in diesem Fall auch die Rechenzeiten zur Ermittlung der Schranken (14) kleiner als die zur Ermittlung der Schranken (11) bzw. (12); denn die mit wachsendem u bzw. v exponentielle Zunahme der Rechenzeit zur Erzeugung der Orthogonalformen fällt bei relativ kleinen u und v noch nicht entscheidend ins Gewicht.
Es seien

$$p_1(u) = \sum_{i=1}^{u} E(G_i) \qquad \text{und} \qquad p_2(v) = 1 - \sum_{i=1}^{v} E(H_i) .$$

Dann ist $\{p_1(u)\}$ ($\{p_2(v)\}$) eine monoton wachsende (fallende) Folge in $u(v)$ mit

$$p_1(u) \leqq p_s \leqq p_2(v), \tag{15}$$

wobei für $u = w$ ($v = s$) der exakte Wert der Systemverfügbarkeit erreicht wird: $p_s = p_1(w) = p_2(s)$. Als Schätzung für p_s kann

$$\hat{p}_s = (p_1(u) + p_2(v))/2$$

dienen. Der maximale Fehler ist kleiner als $d(u, v) = (p_2(v) - p_1(u))/2$:

$$|p_s - \hat{p}_s| \leqq d(u, v); \quad u = 1, 2, \dots, w; \quad v = 1, 2, \dots, s. \tag{16}$$

Wegen $d(w, s) = 0$ können u und v so bestimmt werden, daß der maximale Fehler $d(u, v)$ kleiner oder gleich einem vorgegebenem ε ist: $d(u, v) \leqq \varepsilon$. Dies geschieht durch schrittweise Vergrößerung von u und v, ausgehend von $u = u_0$ bzw. $v = v_0$ bereits ermittelten minimalen Pfad- bzw. Schnittmengen. In jedem Schritt sind eine weitere minimale Pfadmenge W_u und eine minimale Schnittmenge S_v zu erzeugen, die zugehörigen Orthogonalformen für G_u und H_v sind zu konstruieren; $E(G_u)$, $E(H_v)$, $p_1(u)$, $p_2(v)$ und $d(u, v)$ sind zu berechnen. Zur Berechnung von $p_1(u)$ und $p_2(v)$ wird zweckmäßigerweise von den rekursiven Beziehungen

$$p_1(u) = p_1(u - 1) + E(G_u) \quad \text{und} \quad p_2(v) = p_2(v - 1) + E(H_v)$$

Gebrauch gemacht. Natürlich kann man dieses Verfahren so modifizieren, daß u festgehalten und nur v vergrößert wird bzw. umgekehrt. Dieses Vorgehen ist vor allem dann zweckmäßig, wenn die Vergrößerung von $v(u)$ zu einer schnellen Abnahme von $d(u, v)$ führt, während die Zunahme von $u(v)$ den Fehler $d(u, v)$ kaum noch beeinflußt. Das Vorliegen dieser Fälle hängt in erster Linie von den Mächtigkeiten der verwendeten Pfad- bzw. Schnittmengen ab. Ebenso kann es sich als vorteilhaft erweisen, von gleichen Anzahlen u_0 und v_0 minimaler Pfad- und Schnittmengen auszugehen: $r = u_0 = v_0$. In diesem Fall hat man anstelle von (15) und (16)

$$p_1(r) \leqq p_s \leqq p_2(r)$$
$$|p_s - \hat{p}_s| \leqq d(r, r), \qquad r = 1, 2, \dots, \min(w, s).$$

Entscheidend für die Effektivität des Verfahrens ist eine solche Anordnung der minimalen Pfad- und Schnittmengen, daß Mengen mit höherer zuverlässigkeitstheoretischer Importanz vor solchen mit niedrigerer stehen. Diese Importanzen (Wichtigkeiten) hängen zwar auch von den Verfügbarkeiten der in den Pfad- oder Schnittmengen enthaltenen Elemente ab, jedoch haben Pfad- oder Schnittmengen i. allg. eine umso höhere zuverlässigkeitstheoretische Wichtigkeit für die Systemverfügbarkeit, je weniger Elemente sie enthalten [4]. Der anschauliche Hintergrund für diesen Sachverhalt ist der, daß Pfad- und Schnittmengen höherer Mächtigkeit in einer zugehörigen Orthogonalform zu Produkten mit einer höheren Anzahl von Faktoren X_i bzw. $\bar{X}_j$ führen als Pfad- und Schnittmengen kleinerer Mächtigkeit. Bei der Berechnung der Systemverfügbarkeit sind die X_i bzw. $\bar{X}_j$ — Unabhängigkeit der Elemente vorausgesetzt — durch p_i bzw. $\bar{p}_j$ zu ersetzen. Wegen $0 \leqq p_i \leqq 1$ werden somit Produkte mit zunehmender Anzahl von Faktoren in der Tendenz immer kleiner, sie leisten also einen immer kleiner werdenden Beitrag zur Systemverfügbarkeit. Daher empfiehlt es sich, die minimalen Pfad- und Schnittmengen von vornherein nach solchen Verfahren zu erzeugen, die diese Mengen in aufsteigender Mächtigkeit liefern. Für die minimalen Pfadmengen sind dies die „Breitensuchverfahren" (breadth-first search) [2] und [7]. Liegen nicht genügend minimale Pfad- bzw. Schnittmengen mit relativer geringer Mächtigkeit vor, so erweist sich das Näherungsverfahren als ungeeignet. Dieser Sachverhalt trifft vor allem auf die Berechnung unter Schranken für die totale Zusammenhangswahrscheinlichkeit ungerichteter Netzstrukturen vermittels der zugehörigen minimalen Pfadmengen zu, denn in diesem Fall sind die minimalen Pfadmengen gleich den Gerüsten der Netzstruktur, so daß ihre Mächtigkeit stets $m - 1$ ist, wenn m die Anzahl der Knoten der Netzstruktur ist [2].

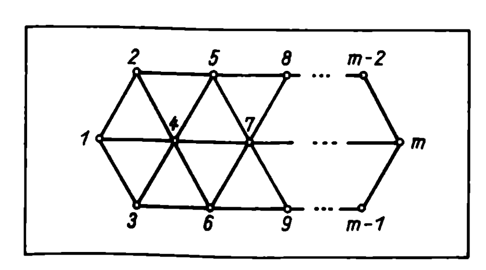
Beispiel 1

Bild 1 zeigt das Zuverlässigkeitsschaltbild eines Systems mit $n = 5 + 7(k - 1)$ Kanten ($k = 1, 2, \dots$) und den Eingangs- bzw. Ausgangsknoten $k_E = 1$ bzw. $k_A = m$. Jedem Element des Systems entspricht genau eine Kante und umgekehrt. Tafel 1 enthält für $m = 4, 7, \dots, 19$ die Anzahlen w bzw. s der minimalen Pfade bzw. Schnitte der zugehörigen monotonen Systeme. Bereits für $m \geqq 16$ ist eine Orthogonalisierung der Strukturfunktion (2) bzw. (3) und damit eine exakte Berechnung der Systemverfügbarkeit auch mit den zur Zeit rechentechnisch effektivsten Verfahren [1] bis [4] und [6] und unter Nutzung moderner Rechner kaum noch möglich. Sie ist jedoch auch nicht notwendig, wie die Anwendung des Näherungsverfahrens auf den Fall $m = 19$ zeigt. Untere Schranken $p_1(u)$ wurden für 43 minimale Pfadmengen und obere Schranken $p_2(v)$ für 15 minimale Schnittmengen berechnet. Die Tafeln 2 und 3 zeigen einen Teil der ermittelten Resultate. Ferner enthalten sie die durch (17) definierte Länge d der zugehörigen Orthogonalform, die auf der Basis des in [4] beschriebenen Verfahrens erreicht wurde. Es ist $p_2(43) - p_1(15) = 0{,}0104$ und somit $d(43{,}15) = 0{,}52\%$. Eine derart geringe maximal mögliche Abweichung der Schätzung $\hat{p}_s = \hat{p}_s(43{,}15) = 0{,}99235$ von p_s fällt in der Praxis schon durch die den Ausgangsdaten anhaftende Unschärfe nicht ins Gewicht. (Die Ergebnisse wurden mit einem Bürocomputer in weniger als einer Minute ermittelt).

Beispiel 2

Bild 2 zeigt eine quadratische Netzstruktur mit $m = 64$ Knoten und $n = 112$ Kanten. Die Knoten sind absolut zuverlässig, die Kanten sind störanfällig und haben die gleiche Verfügbarkeit p. Es interessiert die paarweise Zusammenhangswahrscheinlichkeit $P(s, t)$ der Knoten $s = 30$ und $t = 35$, d. h. die Wahrscheinlichkeit dafür, daß zwischen diesen Knoten ein Weg existiert, der nur aus funktionstüchtigen Kanten besteht [2]. Eine solche Fragestellung interessiert zum Beispiel in Fernmeldenetzen. Zwischen $s = 30$ und $t = 35$ existieren mehr als 100000 Minimalwege und mehr als 10000 Minimalschnitte. Eine exakte Berechnung von $P(s, t)$ durch Orthogonalisierung der zugehörigen Strukturfunktion ist also vollkommen ausgeschlossen. Vermittels des Näherungsverfahrens wurden untere Schranken $p_1(u)$ für $u = 1, 2, \dots, 60$ und obere Schranken $p_2(v)$ für $v = 1, 2, \dots, 26$ ermittelt [7]. Dabei wurden nur Minimalwege mit einer Länge $\leqq 8$ und Schnittmengen der Mächtigkeit $\leqq 10$ verwendet. Die Tafeln 4 und 5 enthalten einige Resultate für $p = 0{,}85$ und $p = 0{,}95$. Man erkennt, daß spätestens für $u = 60$ und $v = 26$ die maximalen Fehler $d(60{,}26)$ der zugehörigen Schätzungen $\hat{p}_s$ vernachlässigbar klein ausfallen. So ist $d(60{,}26) = 0{,}65\%$ für $p = 0{,}85$ und $d(60{,}26) = 0{,}008\%$ für $p = 0{,}95$. Erwartungsgemäß wächst der Fehler mit fallendem p. In Tafel 5 fällt auf, daß trotz

Bild 1. Zuverlässigkeitsschaltbild eines Systems, das aus $n = 5 + 7(k - 1)$ Elementen ($k = 1, 2, \dots$) besteht (Beispiel 1)



Tafel 2. Untere Schranken für p_s (Beispiel 1)

u	d	$p_1(u)$
5	7	0,64528
10	14	0,80694
15	33	0,88167
20	73	0,90396
25	149	0,92104
30	295	0,93983
35	598	0,95629
40	1104	0,97646
43	1618	0,98714

Tafel 3. Obere Schranken für p_s (Beispiel 1)

v	d	$p_2(v)$
2	4	0,99800
4	11	0,99782
8	34	0,99762
12	503	0,99759
15	1828	0,99756

Tafel 4. **Untere Schranken für die paarweise Zusammenhangswahrscheinlichkeit $P(30,35)$ (Beispiel 2)**

u	d	$p = 0,85$	$p = 0,95$
		$p_1(u)$	$p_1(u)$
2	3	0,66686	0,89392
5	11	0,86276	0,98309
10	26	0,90079	0,98964
15	54	0,94712	0,99797
20	133	0,96479	0,99910
30	1122	0,98390	0,99979
60	2510	0,98594	0,99982

Tafel 5. Obere Schranken für die paarweise Zusammenhangswahrscheinlichkeit $P(30,35)$ (Beispiel 2)

v	d	$p = 0,85$	$p = 0,95$
		$p_2(v)$	$p_2(v)$
2	5	0,99899	0,99998
6	29	0,99897	0,99998
13	696	0,99896	0,99998
26	4450	0,99896	0,99998

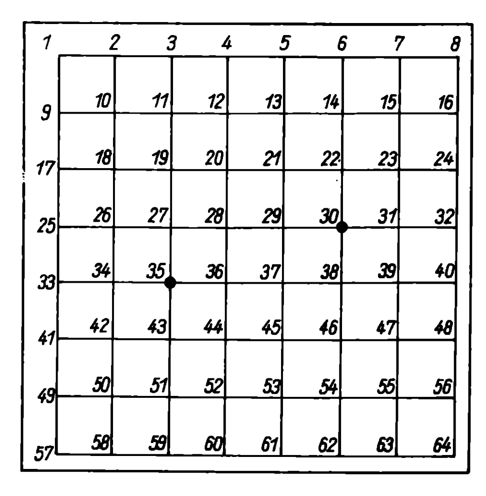


Bild 2. Netzstruktur mit $m = 64$ Knoten und $n = 112$ Kanten (Beispiel 2)

des mit v raschen Anwachsens der Länge der Orthogonalformen die oberen Schranken in den ersten fünf Stellen nach dem Komma nicht beeinflußt werden.

Zusammenfassung

Die Strukturfunktion eines binären monotonen Systems ist prinzipiell durch seine minimalen Pfad- und Schnittmengen gegeben. Die exakte Berechnung der Systemverfügbarkeit durch Ermittlung des Erwartungswertes der Strukturfunktion scheitert jedoch für große, kompliziert vermaschte Systeme an den auch für moderne Rechner unvertretbar hohen Rechenzeiten sowohl für Bestimmung dieser Mengen als auch zur notwendigen Überführung der Strukturfunktion in eine Orthogonalform. Daher haben Näherungsverfahren zur Abschätzung der Systemverfügbarkeit eine große praktische Bedeutung. Das vorgestellte Verfahren benötigt zur Konstruktion von hinreichend genauen Abschätzungen der Systemverfügbarkeit nur einen Bruchteil der minimalen Pfad- und Schnittmengen. Seine rechentechnische Effektivität ist dann gesichert, wenn mit den minimalen Pfad- und Schnittmengen relativ kleiner Mächtigkeit gearbeitet wird und eine genügende Anzahl solcher Mengen vorhanden ist.

Literatur

[1] Abraham, J. A.: An improved algorithm for network reliability. IEEE Trans. Reliab. 28 (1979) 1, S. 58—61.
[2] Beichelt, F.: Zuverlässigkeit strukturierter Systeme. Band 229 der REIHE AUTOMATISIERUNGSTECHNIK. Berlin: VEB Verlag Technik 1988.
[3] Beichelt, F.; Spross, L.: An improved Abraham-method for generating disjoint sums. IEEE Trans. Reliab. 36 (1987) 1, S. 70—74.
[4] Beichelt, F.; Spross, L.: Funktionelle Darstellung der zuverlässigkeitstheoretischen Struktur technischer Systeme durch Orthogonalformen ihrer Strukturfunktion. msr, Berlin 30 (1987) 11, S. 496—499.
[5] Lambert, H. E.: Measures of importance of events and cut sets in fault trees. Reliability and Fault Tree Analysis (eds.: R. E. Barlow, J. B. Fussel, N. D. Singpurwalla). SIAM, Philadelphia 1975.
[6] Programmpaket „Zuverlässigkeitsanalyse binärer monotoner Systeme und stochastischer Netzstrukturen". Ingenieurhochschule Mittweida, Sektion Informationselektronik, Mittweida 1988.
[7] Spross, L.: Effektive Methoden der Zuverlässigkeitsanalyse stochastischer Netzstrukturen. Dissertation B, Hochschule für Verkehrswesen „Friedrich List" Dresden, 1988.

msr 8856

C.-G. Grundig

Zur Methodik der experimentell-analytischen Ermittlung von Gesetzmäßigkeiten diskreter Fertigungsprozesse

0. Einleitung

Die Kenntnisse über innere Wirkungszusammenhänge diskreter Fertigungsprozesse in der mvI als einer wesentlichen Grundlage zu deren gezielter Gestaltung und Steuerung sind unbefriedigend. Ausdruck dieser Situation ist die Begrenztheit der Modellierungsfähigkeit dieser Prozesse, die Vielzahl durchgeführter Simulationsexperimente und die dabei deutlich werdenden Ergebniswidersprüche.
Davon ausgehend werden nachfolgend methodische Schritte einer speziellen Analysestrategie (flächenorientierte Prozeßanalyse) vorgestellt, im Rahmen derer auf Basis

- von getypten und klassifizierten Fertigungsprozessen (Bedienungstheorie)
- eines Experimentieraufbaus nach optimaler Versuchsplanung (SVP) und -auswertung (SVA)
- des Rechnereinsatzes als Simulator (Dialogsysteme CAD)
- der Erstellung eines mathematischen Modells (Prozeßgleichungen)

eine umfassende Testung relevanter Größen als Voraussetzung zur mathematisch-analytischen Modellbildung vorzunehmen ist.
Grundlage für eine gezielte Projektierung (Dimensionierung, Strukturierung, Gestaltung) und Betreibung (Produktionsprozeßsteuerung) effektiver diskreter Fertigungsprozesse einschließlich deren Bestückung mit informationsverarbeitender Gerätetechnik sind gesicherte Kenntnisse über qualitative und quantitative Wirkungszusammenhänge zwischen Einfluß- (Ursachen) und Verhaltensgrößen (Wirkungen). Im günstigen Fall sind diese Wirkungszusammenhänge als mathematisch-analytisches Modell (Verhaltensmodell) in

Dr.-Ing. Claus-Gerold Grundig (44) studierte bis 1968 Technologie des Maschinenbaus an der TH Karl-Marx-Stadt. Danach im VEB Carl Zeiß JENA auf dem Gebiet der Rechnersteuerung von Fertigungsprozessen tätig. Von 1972 bis 1981 wissenschaftlicher Assistent an der TH „Otto von Guericke" Magdeburg. 1977 Promotion A (Simulationsanalyse diskreter Fertigungssysteme). Seit 1981 im VEB GRW „Wilhelm Pieck" Teltow als Fachgebietsverantwortlicher und Gruppenleiter F/E (Aufbau flexibel automatisierter Fertigungsabschnitte). Nebenamtliche Lehrtätigkeit an der TU „Otto v. Guericke" Magdeburg (Produktionsprozeßsteuerung), 1983 Verleihung der Facultas docendi (Betriebsgestaltung).

Form von Gleichungssystemen relevanter Größen formulierbar, aus denen entsprechende Gesetzmäßigkeiten der Fertigungsprozesse exakt ableitbar sind. Deren Kenntnis ermöglicht schon im Projektierungsstadium begründete Vorausberechnungen zu Projektierungs- und Steuergrößen sowie zu Geräteparametern (Leistungsfähigkeit) der Steuerungs- und Automatisierungstechnik und ist damit eine wesentliche Grundlage für eine Optimierung des Prozeßverhaltens. Tendenzen der verstärkten Prozeßautomatisierung folgend, stellt damit die Prozeßmodellierung eine wesentliche Grundlage zur Auswahl und Anpassung von Automatisierungsmitteln der Produktionsprozeßsteuerung und -überwachung entsprechend der in [1] geforderten automatisierungsgerechten Prozeßgestaltung dar.
Wird der gegenwärtige Erkenntnis- und Anwendungsstand zur Prozeßanalyse und Modellbildung betrachtet, so zeigen sich beträchtliche Unterschiede hinsichtlich der teilweise erzielten Analyse- und Modellierungsgenauigkeit einerseits und deren industrieller Anwendungsbreite andererseits.
So wird für spezielle progressive Problembereiche (z. B. Aufbau technologischer Einheiten, integrierte Fertigungsabschnitte) der Einsatz von rechnergestützten Projektierungssystemen, Dialogarbeitsplätzen (insbesondere in wissenschaftlichen Einrichtungen) immer sichtbarer, wobei allerdings festgestellt werden muß, daß damit Vielzahl und inhaltliche Breite betrieblicher Gestaltungs- und Rationalisierungsaufgaben nicht abgedeckt werden können.
Wird weiterhin davon ausgegangen, daß auch perspektivisch

— nicht in jeder betrieblichen Projektierungseinheit flexibel nutzbare dialogorientierte Rechentechnik (Datenbanken, Software) zur Verfügung steht (Auslastung!)
— der Einsatz der Rechentechnik zur Testung des Prozeßverhaltens eine zusätzliche Aktivität im ohnehin zeitlich/kapazitätsseitig angespannten Projektierungsablauf ist
— ein hoher methodisch-inhaltlicher Aufwand zur Realisierung der Simulationsexperimente erforderlich ist (vgl. [19] [26] und [27]),

dann wird allein aus diesen Gründen deutlich, daß zur Qualifizierung der überragenden Vielzahl „konventionell" ablaufender Projektierungsprozesse die nutzerfreundliche Vorgabe von universell anwendbaren Gesetzmäßigkeiten (Normative, Nomogramme, Tabellen, Kennzahlen) zur Qualifizierung der Projektierungsergebnisse erforderlich ist.
Neuere Tendenzen der Prozeßgestaltung [2] bis [5] sind charakterisiert durch

— weitgehende funktionelle, informationelle und räumliche Integration der Prozeßbestandteile bzw. -elemente
— Autonomisierung und Durchgängigkeit der Informationsverarbeitung (CIM-Konzepte)
— Objektivierung der Prozesse (Bedien- und Überwachungsarmut)
— Automatisierung und Dezentralisierung von Abschnitten und Funktionen bei hierarchischer Strukturierung
— Automatisierung komplexer technologischer Abläufe.

Konsequenzen daraus sind ein beträchtliches Ansteigen der Ausrüstungskosten, woraus sich — aus Gründen der Gewährleistung der Prozeßökonomie — zwangsläufig schärfere Forderungen an die Präzision der Prozeßgestaltung und -steuerung ableiten [27].
Damit werden höhere Anforderungen an die detaillierte Kenntnis von Vorgabe- und Ergebnisgrößen gestellt, die nur durch exakte Analyse der Prozeßgesetzmäßigkeiten erbracht werden können.
Nachfolgend werden auf Basis theoretischer und praktischer Untersuchungen Ansatzpunkte zu einer speziellen Methodik der Prozeßanalyse diskreter Fertigungsprozesse bei Nutzung der Simulationstechnik entwickelt, aus denen sich weiterführende Aufgabenstellungen für die Forschung sowie praktische Umsetzung im Rahmen der experimentellen Versuchsgestaltung ergeben.

1. Problemstellung

Bezogen auf den Problemfall „Projektierung und Betreibung kontinuierlicher Prozesse" ist die Erarbeitung mathematischer Modelle unter Einsatz analytischer Methoden zur Identifikation (theoretische/experimentelle Prozeßanalyse) schon seit Jahren üblich. Damit kann im Regelfall der Prozeß durch ein mathematisches Modell hinreichend exakt beschrieben werden. Dieses ist eine wesentliche Grundlage für dessen anlagentechnische Gestaltung einschließlich der Auslegung der Steuer- und Regelprozesse. Im Unterschied dazu zeigt sich bei der Betrachtung des hier vorliegenden Problemfalls, der „Projektierung und Betreibung diskreter, diskontinuierlicher Prozesse" (wie sie im Maschinen- und Gerätebau, in der Bauelementefertigung u. a. typisch sind), daß die Beschreibung des Prozeßverhaltens durch mathematische Modelle im Regelfall nicht bzw. nur bedingt möglich ist.
Hauptgründe liegen im bedienungstheoretischen Grundcharakter realer diskreter Fertigungsprozesse, der allgemein durch *Mehrphasenbedienung technologisch ungleicher Forderungen (Lose) bei stark gemischter Serien-/Parallelstruktur technologisch unterschiedlicher Bedienungskanäle (Ausrüstungen)* charakterisiert werden kann. In [6] wird dafür der Begriff „offener, netzartiger Bedienungsprozeß" geprägt.
Bedienungssysteme dieser Charakteristik entziehen sich der mathematisch-analytischen Modellierung auf „direktem Weg" (im Sinne einer theoretischen Prozeßanalyse). Nur stark vereinfachte Grundfälle sind der direkten mathematischen Modellierung zugänglich.
Auch über Markov-Annahmen konnten aufgrund der hierbei notwendigen Einschränkungen der Ankunfts- und Bedienzeitverteilungen nur sehr begrenzte Modellierungserweiterungen erzielt werden [7] bis [9]. Um trotzdem an praxisnahen existenten bzw. geplanten Prozessen Untersuchungen vornehmen zu können, wurde auf der Basis von Experimenten eine Vielzahl von rechnergestützten Simulationsuntersuchungen durchgeführt.
Dabei wurde auf „indirektem Weg", d. h. über den Umweg des Experiments (im Sinne einer experimentellen Prozeßanalyse), auf Basis eines algorithmischen Modells des Fertigungsprozesses eine Verhaltensanalyse vorgenommen.
Wird der mit diesen Untersuchungen erreichte Erkenntnisstand kritisch analysiert [10] bis [12], ergibt sich folgendes Bild:

— Nahezu alle Simulationsexperimente sind rein problemspezifisch angelegt, haben stark individuelle Modellkonzeptionen und Schwerpunkte der Ergebnisauswertung, d. h. nur wenige Aussagen sind verallgemeinerungswürdig.
— Die Gültigkeitsbereiche der Ergebnisse sind durch ungenügende Beschreibung der Experimentierdaten und des Untersuchungsablaufes nicht fixierbar, auch wird eine detaillierte Beschreibung des Warteschlangensystems als Grundlage zur Charakterisierung des Untersuchungsobjektes (-modelles) nicht vorgenommen.
— Zur Begrenzung des Untersuchungsaufwandes wurden oftmals stark vereinfachende Modellannahmen getroffen, so daß die Abbildungsgenauigkeiten je nach Aussageziel offenbleiben (Verifizierbarkeitsprobleme)
— Den Experimenten wurden nur bedingt statistische Gesichtspunkte zugrunde gelegt und entsprechend ausgewertet, so daß auch dadurch Ergebnisgültigkeit bzw. Gültigkeitsbereiche eingeschränkt sind bzw. oftmals widersprüchliche Ergebnisse auftreten.
— Die Versuchsauswertung wird nicht konsequent bis zur Aufstellung des mathematischen Modells des Prozesses geführt, im Regelfall werden die Meßwerte tabellarisch bzw. grafisch dargestellt und interpretiert. Im Sinne der Prozeßanalyse und -modellierung bleibt man somit auf halbem Wege stehen. Eine Mehrfachausnutzung erkannter Zusammenhänge auf der Basis von Prozeß- bzw. Warteschlangensystemtypen und eindeutiger Zuordenbarkeit von Problemfällen ist bisher nicht möglich.

Die angeführten Probleme erklären die Kluft für die detaillierten Kenntnisse über Prozeßgesetzmäßigkeiten einerseits und die beträchtlichen Grenzen ihrer praktischen Nutzbarkeit andererseits.
Die Tendenz geht deutlich dahin, jeden Problemfall immer wieder neu zu simulieren, obwohl nähere Betrachtungen zeigen, daß, ausgehend von dem allen diskreten Produktionsprozessen in der mvI gemeinsamen Grundcharakter offener bzw. geschlossener Warteschlangensysteme, oft weitgehende Identitäten der Eingangsströme und Bedienungsknoten vorhanden sind.

Damit werden ähnliche bzw. gleiche Fragestellungen und Prozesse immer wieder analysiert, ohne daß signifikante neue Ergebnisse auftreten (z. B. Wirkungsanalyse von Prioritätsregeln auf Bedienungsknoten unterschiedlicher Größe und Struktur) [19].

Es liegt daher nahe, die Simulationsuntersuchungen so aufzubauen, daß umfassende und statistisch abgesicherte Ergebnisse für klassifizierte Prozeßtypen ableitbar sind, die weitere Simulationsexperimente überflüssig machen, sofern Zielsetzungen und Prozeß vergleichbar sind. Es erscheint sinnvoll, Methodik und Zielsetzung der Analyse diskreter Fertigungsprozesse zwei unterschiedlichen Strategien unterzuordnen, die sich hinsichtlich Einsatzzweck, Untersuchungsablauf, Ausgabereiche und Ergebnisform unterscheiden (Tafel 1). Merkmale der Analysestrategien sind:

- **Punktuelle Prozeßanalyse (pPA)**
  erfolgt auf der Basis flexibler dialogorientierter Soft- und Hardware; wird angewendet zur detaillierten Analyse spezieller Prozesse; statistische Anforderungen begrenzt; Aussageziele nur auf Problemfall gerichtet; ohne Zielsetzung der Verallgemeinerungsfähigkeit; mathematisches Modell wird nicht explizit ermittelt
- **Flächenorientierte Prozeßanalyse (fPA)**
  erfolgt auf der Basis flexibler dialogorientierter Soft- und Hardware; wird ausschließlich angewendet für Grundsatzanalysen getypter und klassifizierter Prozesse; statistische Anforderungen sehr hoch (Einsatzmethodik statistische Versuchsplanung); mathematisches Modell ist zu erstellen; verallgemeinerungsfähige Aussagen (Normative, Kennzahlen, Nomogramme) bei Absicherung der Problemzuordenbarkeit sind abzuleiten.

Wie deutlich wird, ist die überragende Vielzahl bisher bekannt gewordener Simulationsexperimente in Methodik und Zielsetzung der pPA zuzuordnen, so daß in der weiteren Forschungsarbeit insbesondere die Vorgehensweisen der fPA zu entwickeln sind.

Erste Ansätze zur Simulationsanalyse im Sinne der fPA wurden in [15] gemacht, jedoch stark begrenzt auf den dort zugrunde gelegten Anwendungsfall.

Perspektivisch stellt die pPA folglich eine problemorientierte Analysemethode bei begrenztem Aussagebereich dar, wohingegen die fPA prozeßbezogen im Rahmen von Grundsatzuntersuchungen mit den Zielsetzungen breiter Anwendbarkeit der Aussageergebnisse ohne ständig wiederkehrenden Experimentieraufwand zu betreiben ist. Methodisch gesehen wird dabei nach [13] und [14] der Fertigungsprozeß von außen her mit der Zielsetzung der experimentellen Modellbildung analysiert.

Ausgehend von der dargestellten Notwendigkeit der Qualifizierung der Simulationsuntersuchungen hinsichtlich Aussagequalität und Nutzungsmöglichkeiten beinhaltet die fPA folgende weitergehende Zielsetzungen:

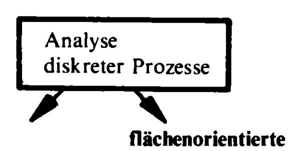
- umfassende Durchtestung des Prozeßverhaltens auf Basis getypter und klassifizierter diskreter Prozesse bezüglich aller relevanten Einfluß- und Verhaltensgrößen
- Modellbildung:
  Untersuchungsobjekt — Warteschlangensystem
  Experimentierablauf — statistische Versuchsplanung
- Umsetzung der Simulationsergebnisse in ein mathematisches Modell (Modellsystem) in Form von Prozeßgleichungen
- Ableitung von statistisch abgesicherten Gesetzmäßigkeiten, Wirkungszusammenhängen, Wirksamkeit von relevanten Größen und deren nutzergerechten Darstellung (als Tabelle, Kennzahlen, Nomogramme u. a.) bezogen auf Prozeßtypen.

Im Ergebnis der fPA wird angestrebt:

- modellseitig und statistische umfassende Durchtestung des Prozeßverhaltens bei Ableitung
  - der Wirksamkeit je Einflußgröße
  - der dominanten bzw. nicht wirksamen Einflußgrößen
  - der komplexen Wirkungszusammenhänge (Gesetzmäßigkeiten)
  - von Kompensationseffekten

  Zielsetzungen sind die Identifizierung kausaler Zusammenhänge (Steuereinwirkungen/Steuerziel) entsprechend den Forderungen in [5] und [16] bis [19].

Tafel 1. Grundstrategien der Analyse diskreter Prozesse



punktuelle Prozeßanalyse (pPA)	**flächenorientierte Prozeßanalyse (fPA)**
Merkmale:	*Merkmale*:
• spezieller Prozeß	• Prozeßtypen, klassifiziert
• Simulation (interaktive EDV-Anlagen)	• Simulation (interaktive EDV-Anlagen)
• Experimentierstrategie wählbar	• vorgegebene Experimentierstrategie (statistische Versuchsplanung)
• Aussagen nur gültig für Untersuchungsobjekt (punktueller Aussagebereich)	• Aussagen gültig für alle den Prozeßtypen zuordenbare Problemfälle (flächenorientierter Aussagebereich)
• Ergebnisse — Tabellen — Graphiken — Histogramme	• Ergebnisse — mathematisches Modell (Prozeßgleichungen) — Nomogramme, Tabellen, Funktionen

- Nutzung der abgeleiteten Gesetzmäßigkeiten aus der fPA zur Projektierung von Gestaltungs- und Steuerungslösungen auf Basis der Gewährleistung der Zuordnung Problemtyp zu Prozeßtyp
- Einsatz pPA folglich nur noch für spezielle Untersuchungen solcher Prozesse, die sich der Prozeßtypung als Basis der fPA entziehen.

Nachfolgende Ausführungen beziehen sich ausschließlich auf spezielle Aspekte der Konzeption einer fPA.

2. Untersuchungsgrundlagen

Grundlage zur mathematisch-analytischen Modellierung stochastischer Fertigungsprozesse sind die Formalisierung von relevanten Einfluß- und Verhaltensgrößen (Zielgrößen) sowie deren Wechselbeziehungen. In allgemeiner Form ergibt sich der Grundzusammenhang zwischen Einflußgröße x (unabhängige Variable) und Verhaltensgröße y (abhängige Variable) zu

$$y = T(x) . \tag{1}$$

Das Kernproblem besteht in der Formulierung des Übertragungsoperators T, d. h., die Wirkungszusammenhänge zwischen x_i und y_i sind zu bestimmen.

Den Größen x_i kann Zufallscharakter zugeordnet werden, so daß auch deren Realisierungen y_i Zufallsgrößen sind, auch können scharfe bzw. unscharfe Annahmen getroffen werden.

Das gesuchte Ergebnis der fPA (mathematisches Modell) wird je nach Komplexität des Problemfalles durch ein System von Regressionsgleichungen (Prozeßgleichungen) gebildet. Für die hier betrachteten Fälle ist ein System von multiplen Regressionsgleichungen 2. Grades zu erwarten (vgl. auch [17] und [18]), so daß folgende allgemeine Modellannahme angesetzt werden kann (mathematische Struktur der Wirkungsfläche):

$$y = b_0 + \sum_{i=1}^{k} b_i x_i + \sum_{i=1}^{k} \sum_{j=1}^{k} b_{ij} x_i x_j . \tag{2}$$

Damit sind sowohl direkte (lineare und quadratische) Effekte der Einflußgrößen als auch paarweise Wechselwirkungen zwischen ihnen faßbar. Die Bestimmung der Parameter des Ansatzes erfolgt durch Simulationsexperimente auf Basis von zentral zusammengesetzten Versuchsplänen entsprechend den Methoden der statistischen Versuchsplanung (Einsatz statistischer Experimentplanungs- und Auswertungsprinzipien).

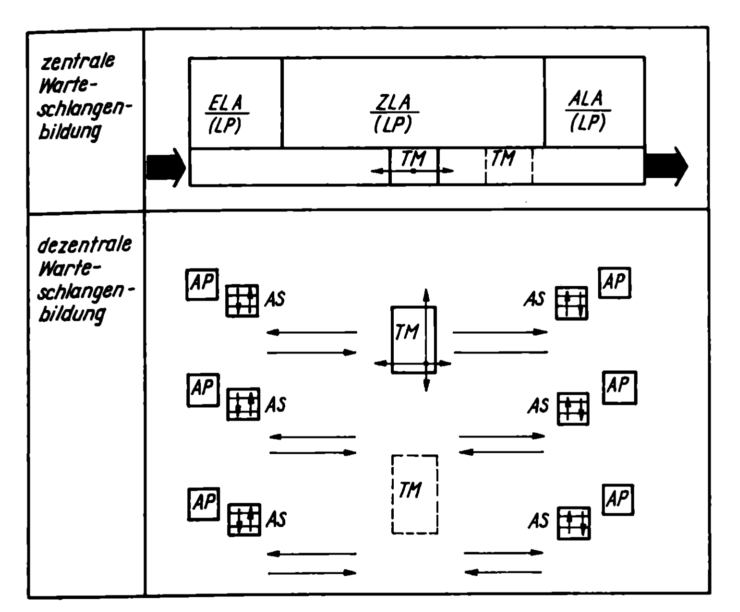


Bild 1. Hierarchie der Warteschlagenbildung (FL/TM/AP/LP) in der Prozeßstufe „Teilefertigung" (schematisches Grundprinzip)
FL Fertigungslose; *TM* Transportmittel; *AP* Arbeitsplatz; *LP* Lagerplatz; *AS* Arbeitsplatzspeicher; *ELA* Eingangslagerung; *ZLA* Zwischenlagerung; *ALA* Ausgangslagerung

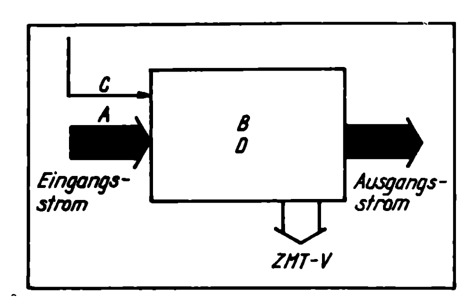


Bild 2. Hauptkomponenten der Warteschlangenbeeinflussung im offenen diskreten Fertigungsprozeß

Tafel 2. Komponenten der Prozeßbeeinflussung

Stadium der Prozeßbeeinflussung	Einstellbare Einflußgrößen			
	A	*B*	*C*	*D*
I. Projektierungsphase • Funktionsbestimmung • Dimensionierung • Strukturierung • Gestaltung		×		×
II. Durchführungsphase • mittel-/kurzfristige Produktionsplanung • Produktionsprozeßsteuerung	×		×	×

Tafel 3. Verhaltensgrößen des ZMT-V

Zeitverhalten (ZV)	— Auslastungsgrade • Arbeitsplätze • Transportmittel (zentral/dezentral) • Lagermittel (zentral/dezentral) — Durchlaufzeiten • Fertigungsaufträge — Liegezeiten • Fertigungsaufträge — Zeitaufwände für TUL-Prozeß
Mengenverhalten (MV)	— Warteschlangenbildung der Aufträge • vor Arbeitsplätzen (-gruppen) • in Lagersystem (zentral/dezentral) — Warteschlangenbildung der Arbeitsplätze — Warteschlagenbildung der Transportmittel
Terminverhalten (TV)	— Grad der Termineinhaltungen • auftragsbezogen • bereichsbezogen — Grad der Terminverzüge (Anzahl, Größe, Zeitpunkte) • Hauptprozeß • Hilfsprozeß

Einflußgrößen x_i

Werden die zu definierenden Einflußgrößen betrachtet, so zeigt sich die Komplexität des Modells. Es sind deshalb Beschränkungen auf signifikante Größen erforderlich.
In [20] wird von 10^3 bis 10^6 Einflußfaktoren ausgegangen. Es handelt sich dabei um die Größen, durch die die zentrale/dezentrale Warteschlangenbildung von Werkstücken (Losen) bzw. von Ausrüstungen (Arbeitsplätze, Lagerplätze, Transportmittel) beeinflußt wird und die damit Wirkungen auf Verhaltensgrößen haben (Bild 1).
Mit der bedienungstheoretischen Betrachtungsweise des Fertigungsprozesses gemäß Bild 2 können Einflußgrößengruppen *A*, *B*, *C*, *D* als Hauptkomponenten der Warteschlangenbeeinflussung gruppiert werden: *A* Eingangsstrom der Werkstücke; *B* Bedienungsknoten; *C* Organisation Warteschlangenabbau; *D* Störungen.
Eine Vielzahl von Simulationsuntersuchungen ergab, daß durch das Zusammenwirken von *A*, *B*, *C* und die Überlagerung von *D* signifikante wechselseitige Wirkungen bestehen, die die Verhaltensgrößen y_i maßgeblich beeinflussen [21].
Damit sind Ansatzpunkte zur gezielten Prozeßbeeinflussung in den Stadien der Projektierung bzw. Produktionsdurchführung durch spezielle Maßnahmen (z. B. Struktur Bearbeitungssystem, Methode der PPS) gegeben, die auf einer weitgehenden Einstellbarkeit der Komplexe *A*, *B*, *C* und *D* einerseits und einer Konstituierbarkeit der Verhaltensgrößen entsprechend den Zielfunktionen andererseits beruhen. In Tafel 2 wird deutlich, daß Phase I (Projektierung der Fertigungsstätte), ausgehend von der Produktionsaufgabe, insbesondere durch die direkt bzw. indirekt zu fixierenden Größen Kapazität (Anzahl), Struktur (Kopplung) und technologische Flexibilität (Ausrüstungstyp) sowie die einzubauenden Reserven (Rundungen, Redundanzen) über die sich in Phase II einstellende Warteschlangenbildung (Staubildung) vorab entscheidet.
Das Projektierungsergebnis der Phase I hat damit nahezu statischen Charakter und kann in Phase II unter dem Aspekt der Beeinflussung der Warteschlangenbildung nur bedingt korrigiert werden (Einflußgrößengruppe *B* ist quasistatisch).
In Phase II (Produktionsplanung, Produktionsprozeßsteuerung) wird die Warteschlangenbildung durch die Methodik der Planeinordnung (Zeitraum), Einschleusung in den Fertigungsabschnitt (Gruppe, Reihenfolge), Strategie des Warteschlangenabbaues (Prioritätsregeln) sowie der Störungskompensation (Verlagerungen, Ausweichplätze) beeinflußbar. Die Ergebnisse der Phase II tragen dynamischen Charakter, ermöglichen damit eine ständig gezielt veränderbare Warteschlangenbeeinflussung.

Verhaltensgrößen y_i

Die Verhaltensgrößen (Tafel 3) charakterisieren das zeitabhängige dynamische Verhalten der Elemente im Haupt- und im Hilfsprozeß und werden durch deren Zeit-, Mengen- und Terminverhalten (ZMT-V) definiert [11].
Sind im Ergebnis der (prozeßtypbezogenen) fPA wirkungsrelevante Einflußfaktoren, deren Wirkungscharakteristik, Wirksamkeit sowie Wirkungszusammenhänge bekannt, sind, davon ausgehend, Optimierungsstrategien des Prozeßverhaltens auf Basis bekannter Grundzusammenhänge durchzutesten und zu formulieren.
Bisherige Simulationsuntersuchungen zum ZMT-V zeigen z. B. folgende verallgemeinerungsfähige Verhaltenszusammenhänge [21] [22] und [15]:

— Steigende Einschleusungen (Lose) je Zeitintervall führen zum Ansteigen der Warteschlangenbildung (Auslastungsgrade, Durchlaufzeiten) einerseits bei gleichzeitigem Absinken der Termineinhaltungen.
— Produktionssysteme erreichen nur quasistationäre Systemzustände durch die Notwendigkeit des Einhaltens der Bedingung $s \cdot \mu > \lambda$ (bedienungstheoretische Bedingung: s Anzahl der Bedienungskanäle, λ Ankunftsrate, μ Abfertigungsrate).

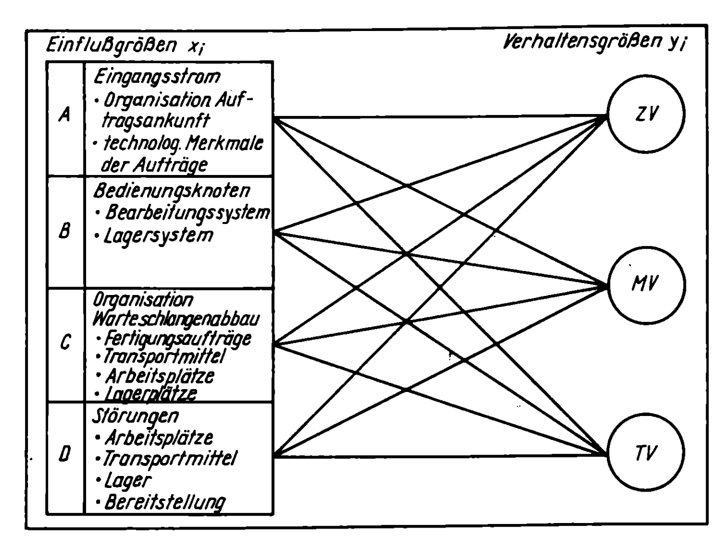


Bild 3. Komplexität des Wirkungszusammenhanges von Einfluß- und Verhaltensgrößen

- Redundanzen im Produktionsprozeß führen zur Beschleunigung im Warteschlangenabbau, d. h. z. B. ersetzende Strukturen des Bedienungsknotens, ungezonte Lagerungen usw. sind anzustreben.
- Die Warteschlagenbildung im Prozeß (Zwischenlagerbestand) wird maßgeblich beeinflußt von der Anzahl der Arbeitsgänge, den Maschinenbelegungszeiten sowie vom Ankunftsabstand der Aufträge.
- Auslastungsveränderungen folgen nur gedämpft den Belastungsveränderungen.
- Verhaltensdominante Größen auf das ZMT-V sind:
  - Ankunftsabstand (Einschleusungsintensität)
  - Arbeitsganganzahl (je Fertigungsauftrag)
  - Belegungsdauer (je Arbeitsplatz)
  - Prioritätsregel (Warteschlangenabbau)
  - Struktur des Bedienungsknotens.

Die gezielte experimentelle Bestimmung von Verhaltensgrößen (gemäß Tafel 3) ermöglicht die Beurteilung des komplexen Systemverhaltens je Prozeßtyp gemäß den Zielsetzungen der fPA.
Die Präzisierung von (1) führt folglich auf Basis der inhaltlichen Definition von x_i und y_i zu

$$(\text{ZMT-V}) = f(A, B, C, D)\,, \tag{3}$$

womit die Komplexität des Untersuchungsfeldes umrissen ist (Bild 3).

3. Systematik der Untersuchungsmethode

Ausgehend von der Notwendigkeit der Formulierung methodisch-inhaltlich abgrenzbarer Teilschritte (TS) sowie deren funktionell bedingter Ablauffolge sind diese im Bild 4 in Form eines Ablaufplanes für die Grundprinzipien der fPA aus Abschn. 1. dargestellt. Nachfolgend wird eine inhaltliche Präzisierung der Teilschritte vorgenommen.
Methodisch wird dabei von folgenden Komplexen ausgegangen (Bild 4):

Komplex I
TS 1 bis TS 5
Entscheidungsfindung zur Methodik der Modellierung
- fPA (ohne mathematisches Modell)
- fPA (mit mathematischem Modell)

Komplex II
TS 6.1 bis TS 6.4
Statistische Versuchsplanung (SVP)

Komplex III
TS 7
Versuchsdurchführung (Simulationsexperimente)

Komplex IV
TS 8 bis TS 10
Statistische Versuchsauswertung (SVA).

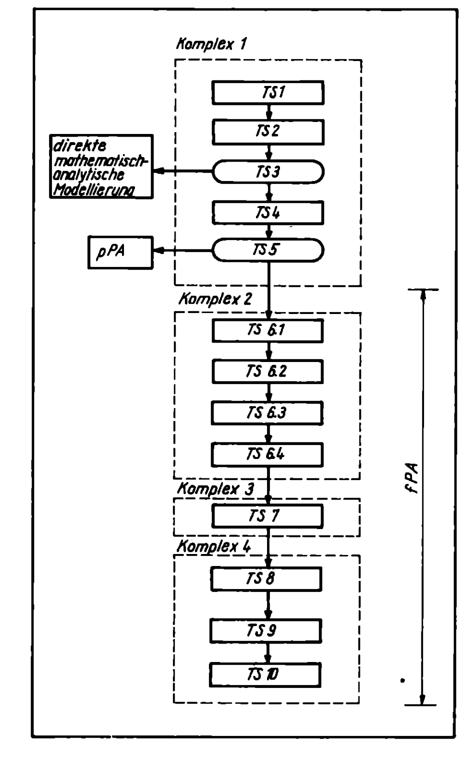


Bild 4. Systematik der Teilschritte der flächenorientierten Prozeßanalyse (fPA)

Zielsetzung der fPA gemäß den Komplexen I bis IV ist damit die experimentelle Ermittlung von Gesetzmäßigkeiten in bisher unzureichend durchdrungenen Forschungsgebieten — Analyse diskreter Fertigungsprozesse — basierend auf zielgerichteten statistischen Planungs- und Auswertungsprinzipien der Simulationsexperimente [17].

Präzisierung der Teilschritte:

Teilschritt 1
- Erarbeitung und Präzisierung des Untersuchungszieles
- Definition der zu untersuchenden Zielgrößen und Optimierungsbereiche
- Erfassung relevanter Einflußgrößen
- Entwicklung Grobmodell des Prozesses gemäß Problemfall auf bedienungstheoretischer Grundlage

Teilschritt 2
Einordnung des Grobmodells in bedienungstheoretische Grundmodelle und Prüfung auf mathematisch-analytische Modellierbarkeit auf direktem Weg

Teilschritt 3 (Entscheidungsschritt)
Prüfung der Entscheidungskriterien:
- Struktur der Bedienungskanäle
  (Regelfall: gemischte Reihen-/Parallelschaltung)
- Ankunfts- und Bedienungsprozesse
  (Verteilungsfunktionen)
  Regelfall: nicht als Markov-Prozesse einordenbar

Damit scheidet eine detaillierte Modellierung auf rein mathematisch-analytischem Wege aus; es ist die Modellierung über ein algorithmisches Simulationsmodell erforderlich (TS 4)

Teilschritt 4
Zuordnung des Problemtyps (Grundmodell) zu getypten, klassifizierten bedienungstheoretisch formulierten Prozeßgruppen der fPA

Teilschritt 5 (Entscheidungsschritt)
Prüfung auf Zuordenbarkeit
- positiv — fPA ist vorzunehmen
- negativ — pPA ist vorzunehmen

Teilschritt 6.1
Hypothesen zur Struktur von Wirkungszusammenhängen
(Bildung Modellansatz)
- Funktionelle Zusammenhänge Einfluß- und Verhaltensgrößen (Struktur der Funktion T in (1) noch unbekannt aber Existenz von Zusammenhängen wird angenommen)

- Aufstellen von Hypothesen zu wahrscheinlichen Abhängigkeiten von Einfluß- und Verhaltensgrößen (empirisches Kausal- bzw. Abhängigkeitsgefüge); Erfahrungen sind erforderlich
- Formulierung des Systems von Regressionsgleichungen gemäß dem empirischen Abhängigkeitsgefüge (Mathematische Grundstruktur der Wirkungsfläche)

Teilschritt 6.2

Formulierung des Testraums

- Vorgabe von Variations-(Experimentier-)bereichen und Wertekombinationen von Einflußfaktoren
- Festlegen des Experimentiergebietes (Testraum), in dem Wirkungscharakter, Wirksamkeit und Wirkungszusammenhänge der Einflußgrößen und Verhaltensgrößen (TS 6.1) experimentell zu untersuchen sind

Teilschritt 6.3

Entwurf der Wirkungsflächen

- Überlegungen zur wahrscheinlichen Struktur der Wirkungsfläche
- *Möglichkeiten*:
  a) alle Einflußfaktoren wirken voneinander unabhängig auf Verhaltensgrößen
  b) Unabhängigkeit der Einflüsse nicht vorhanden
- Regelfall ist b), d. h. Wechselwirkungen sind vorhanden, damit ist vollständiges Polynom 2. Grades gemäß (2) zur Berücksichtigung linearer, quadratischer und paarweiser Wirkungen anzunehmen. Die mathematische Struktur der Wirkungsfläche ist damit entworfen.

Teilschritt 6.4

Optimale Versuchsplanung (OVP)

- Aufstellen Experimentierprogramm zur Bestimmung der Parameter der Wirkungsflächen (Polynom) bei hoher statistischer Sicherheit und minimalem Experimentieraufwand (Rechenzeiten)
- Festlegen optimaler Versuchspunkte (Wertekombinationen von Einflußgrößen)
- Regelfall: Anzahl der Versuchspunkte $R > 2$; Wirkungsfläche stellt eine Hyperfläche dar
- Entwurf optimaler Versuchspläne

Teilschritt 7

Versuchsdurchführung

- Erarbeitung eines komplexen Simulationsmodells (Prozeßmodell/Experimentiermodell)
- Durchführung der Simulationsexperimente auf EDV-Anlage

Teilschritt 8

Statistische Versuchsauswertung (SVA)

Das methodisch-inhaltliche Vorgehen der SVA ist auf Basis sachbezogener logischer Aspekte vorzunehmen.
Folgende Prüfungen des Modellansatzes sind einzuleiten:

- Modellreduktion
- Modellkorrektur
- Auflösung vermengter Effekte

Teilschritt 9

Aufstellen des mathematischen Modells

- Entwurf eines Systems von Regressionsgleichungen (Prozeßgleichungen) für relevante Einflüsse und Zusammenhänge

Teilschritt 10

Ableitung der Prozeßgesetzmäßigkeiten (Umsetzung TS 9)

Berechnung von wesentlichen Projektierungs- und Steuerungsgrößen auf Basis von TS 9 für prozeßtypische Wertebereiche; nutzergerechte Umsetzung in

- Nomogramme
- Graphiken
- Tabellen
- Kennzahlen
- Richtwerte.

Zusammenfassung

Ausgehend von der Notwendigkeit der systematischen und detaillierten Durchdringung von Wirkungen und Zusammenhängen komplexer diskreter Fertigungsprozesse wurde ein methodisches Konzept zur flächenorientierten Prozeßanalyse (fPA) in seinen Grundzügen entwickelt. Ziel ist es, im Rahmen gezielter Grundlagenforschungen prozeßtypenbezogen Gesetzmäßigkeiten diskreter Fertigungsprozesse unter Nutzung der Simulationstechnik bei Zugrundelegung der Methoden der statistischen Versuchsplanung und -auswertung umfassend aufzudecken und über die dann mögliche Bildung von Prozeßgleichungen mathematisch-analytisch berechenbar zu machen.
Als Schwerpunkte der weiteren Grundlagenforschung zur dargestellten Experimentierstrategie werden angesehen

- die bedienungstheoretisch orientierte Prozeßtypung zur Gewährleistung der Zuordenbarkeit Problemfall/Prozeßtyp
- gezielter Einsatz optimaler Versuchspläne auf Basis fundierter wissenschaftlicher Hypothesenbildung (hypothetische Grundannahmen zur Modellgestaltung auf der Basis von Erfahrungen)
- Rationalität in der Versuchsdurchführung und -auswertung und nutzergerechten Ergebnisumsetzung.

Literatur

[1] Töpfer, H.: Automatisierungsgerechte Prozeßgestaltung — eine Aufgabe der Technologen. Wiss. Zeitschrift der TU Dresden 30 (1981) 2/3, S. 161—164.
[2] Woithe, G.; Gottschalk, E.: Entwicklungstendenzen der Strukturbildung von Werkstätten. Wiss. Zeitschrift der TH Magdeburg 24 (1980) 1, S. 89—93.
[3] Gottschalk, E.; Schreiber, H.: Zum Niveau der Produktionsprozeßsteuerlösung bei bedienarmer Fertigung. Wiss. Zeitschrift der TH Magdeburg 26 (1982) 5, S. 103—108.
[4] Henning, D.; Trognitz, V.; Woithe, G.: Stand und Entwicklungstendenzen der Fertigungsstrukturbildung im Maschinenbau. Fertigungstechnik und Betrieb, Berlin 33 (1983) 8, S. 455—458.
[5] Nasko, H.: Automation: Chancen und Probleme aus technischer Sicht. Haus der Technik Essen. Vortragsveröffentlichung 458.
[6] Kluge, P.-D.; Runge, W.: Zufallsabhängige Fertigungsprozesse. Berlin: Verlag Die Wirtschaft 1984.
[7] Walter, H.-P.: Ein Ansatz für die mathematische Modellierung elementarisierter Fertigungsprozesse auf der Grundlage von Zustandsgleichungen mit Markov-Charakter. Dissertation A, Humbold-Universität zu Berlin 1976.
[8] Eichhorn, N.: Ein Beitrag zur Aufbereitung und Weiterentwicklung von Grundlagen für die stochastische Modellierung diskontinuierlicher Fertigungsprozesse mittels Bedientheorie und Simulation. Dissertation A, Humbold-Universität zu Berlin 1977.
[9] Scheibner, K.: Beitrag zur Modellierung diskontinuierlicher technologischer Prozesse mit Hilfe der Bedienungs- und Automatentheorie. Dissertation A, TU Dresden 1973.
[10] Wegner, W.; Heinemeyer, W.: Einsatz der Simulationstechnik im Produktionsbereich. Fortschrittliche Betriebsführung und Industrial Engineering 25 (1976) 4, S. 225—233.
[11] Grundig, C.-G.: Einsatz der Simulationsmethode zur Verhaltensanalyse von Fertigungsprozessen — Stand, Probleme und Tendenzen. Wiss. Zeitschrift der TH Magdeburg 24 (1980) 2, S. 89—97.
[12] Kreutzfeld, H.-F.: Analyse der Einflußgrößen auf die Terminplanung bei Werkstättenfertigung. Dissertation, TU Hannover 1977.
[13] Töpfer, H.; Bischoff, H.: Grundlagen der Prozeßanalyse. Lehrbrief 1. WGMA in der KDT 1979.
[14] Töpfer, H.; Rieger, P.: Analyse von Prozessen unter Nutzung experimenteller Mittel. Lehrbrief 3. WGMA in der KDT 1980.
[15] Moron, D.: SIGFA — Ein Verfahren zur prozeßnahen Dimensionierung integrierter Fertigungsabschnitte mit Zentralspeicher. Dissertation A, TH Karl-Marx-Stadt 1981.
[16] Klett, E.: Untersuchungen zur Bestimmung mathematischer Modellansätze bei Regressionsanalysen. msr, Berlin 24 (1981) 11, S. 630—633.
[17] Pegel, B.: Empirische Modellbildung und Versuchsplanung. Beiträge zur Forschungstechnologie, Heft 7. Berlin: Akademie-Verlag 1980.
[18] Karl, P.: Untersuchungen zur Anwendung der statistischen Versuchsplanung bei der digitalen stochastischen Simulation diskreter Prozesse. Dissertation A, Ingenieurhochschule Dresden 1979.

(Fortsetzung auf S. 278)

J. Lunze

Wissensbasierte Beratung beim rechnergestützten Entwurf von Automatisierungssystemen

Teil 2: Implementierung des Beratungssystems mit PROLOG[1)]

9. Grundlagen und Syntax von micro-PROLOG

Die Programmiersprache PROLOG ist in der letzten Zeit ausführlich beschrieben worden, so in [4] und [10] bzw. in [2] und [3]. Hier sollen nur einige Hinweise auf micro-PROLOG gegeben werden, um die nachfolgenden Ausführungen auch für Nicht-PROLOG-Programmierer verständlich zu machen.
Grundlage der Programmiersprache PROLOG ist die Prädikatenlogik der ersten Stufe. Ein k-stelliges Prädikat $P(X_1X_2 \dots X_k)$ beschreibt eine Beziehung zwischen k Objekten. Es besteht aus einem Prädikatensymbol P und den Argumenten X_i und wird in der micro-PROLOG-Syntax in der Form $(P\ X_1 \dots X_k)$ geschrieben. Wird für die Argumente X_i die Bezeichnung bestimmter Objekte A_i eingesetzt, so ist $(P\ A_1 \dots A_k)$ eine Aussage, die den Wahrheitswert „wahr" oder „falsch" haben kann. Zum Beispiel ist die Aussage „Die Entwurfsschritt-Nummer ist 1" bzw. (Entwurfsschritt-Nummer 1) zu Beginn des Entwurfes wahr.
In PROLOG werden nur logische Ausdrücke der Form

$$B \Leftarrow A_1, A_2, \dots, A_n \tag{23}$$

zugelassen (sogenannte Horn-Klauseln, vgl. (14)). Dabei bezeichnen B und $A_1, \dots, A_n$ Prädikate, „," die Konjunktion und „$\Leftarrow$" die Implikation. Gleichung (23) kann gelesen werden als „Damit B gilt, müssen A_1 und A_2 und ... A_n gelten". In micro-PROLOG-Syntax wird (23) als Liste

$$(B\ A_1A_2 \dots A_n) \tag{24}$$

notiert. Für $n = 0$ geht (24) in den unbedingten Ausdruck

$$(B) \tag{25}$$

über, der eine Tatsache (Fakt) beschreibt. Die oben angeführte Aussage kann also als unbedingter Ausdruck

((Entwurfsschritt-Nummer 1))

aufgeschrieben werden.
X, Y, Z, x, y, z, X1, Y1 usw. sind Variable, deren Bedeutung jeweils nur für *eine* Klausel der Form (24) festgelegt ist. Sie können sowohl für einzelne Elemente als auch für Listen stehen. Da die Variablen immer in der hier angegebenen Reihenfolge verwendet werden, hängt ihre inhaltliche Bedeutung von den Positionen ab, in denen sie innerhalb eines Prädikates stehen. Die inhaltliche Bedeutung der Variablen wird im folgenden in Definitionsklauseln in Stichworten angegeben.
Listen können durch die Schreibweise (X | Y) in das erste Element X und die Restliste Y zerlegt werden. Beispiele für eine Implikation (24) sind die Definitionen der Prädikate „Element-der-Menge" und „haben-gemeinsames-Element"

```
((Element-der-Menge X (X | Z)))          Element * Liste
((Element-der-Menge X (Y | Z))
  (Element-der-Menge X Z))
((haben-gemeinsames-Element X Y)         Liste * Liste
  (Element-der-Menge Z X)
  (Element-der-Menge Z Y))                               (26)
```

Die erste Klausel besagt, daß das Element X zur Liste (X | Y) gehört. In der zweiten Klausel steht, daß X Element der Liste (Y | Z) ist, wenn X zur Restliste Z gehört. Damit wird das Problem, X als Element der Gesamtliste (Y | Z) nachzuweisen, auf das gleichartige Problem zurückgeführt, X als Element der Restliste Z zu erkennen, wobei für diese Aufgabe rekursiv wieder die beiden Klauseln (26) angewendet werden können. Zwei Listen X und Y haben ein gemeinsames Element Z, wenn Z mit X und Y das Prädikat „Element-der-Menge" erfüllt.
Der Arbeitsweise des PROLOG-Interpreters liegt folgende prozedurale Interpretation der Implikation (24) zugrunde: „Um das Ziel B zu erfüllen, müssen die Teilziele $A_1, A_2, \dots, A_n$ nacheinander erfüllt werden." Der Interpreter sucht deshalb nach einer Klausel (26), deren erstes Prädikat A_1 heißt. Ist dies ein unbedingter Ausdruck (25), so ist das Teilziel A_1 erfüllt. Andernfalls muß versucht werden, die in der entsprechenden Klausel stehenden Teilziele zu erfüllen, usw.
Zur Erfüllung der Teilziele wird das PROLOG-Programm von oben nach unten durchsucht, so daß bei mehreren Klauseln mit demselben Kopf zunächst die oberste — und wenn dies nicht erfolgreich ist, die nachfolgende — verwendet wird. Alle für dieses Vorgehen notwendigen programmtechnischen Vorkehrungen sind im PROLOG-Interpreter realisiert. Gleiches gilt für die beim Vorhandensein von Variablen notwendigen Maßnahmen für das Binden gleichnamiger Variabler innerhalb einer Klausel an eine andere Variable bzw. Konstante.
Der PROLOG-Interpreter verfügt über fest definierte Prädikate (Built-In-Prädikate), von denen hier folgende verwendet werden:

(NOT A)	(Negation) ist erfüllt, wenn die Aussage A nicht gilt
(SUM I_1 I_2 I_3)	(Summenbildung) ist erfüllt, wenn $I_1 + I_2 = I_3$ gilt
(ISALL X Y Z)	ist erfüllt, wenn X die Liste aller Lösungen Y der Zielklausel Z darstellt
(PP X), (P X)	(Ausdruck) ist erfüllt, wenn die Liste X ausgedruckt ist, wobei für Variable die an diese gebundenen Konstanten ausgegeben werden
(R X)	(Einlesen) ist erfüllt, wenn X an eine Terminaleingabe gebunden ist
(KILL X)	(Löschen einer Klausel) ist erfüllt, wenn alle Klauseln für das Prädikat X gelöscht sind
(ADDCL X Y)	(Eintragen einer Klausel) ist erfüllt, wenn die Klausel X nach der Y-ten Klausel gleichen Namens in das PROLOG-Programm eingetragen ist.

Logisches Programmieren bedeutet das Aufschreiben von Wissen über die Problemstellung in Form von Horn-Klauseln (24). Das Programm besteht aus

- **Fakten,** die die Existenz von Objekten bzw. das Vorhandensein von Eigenschaften zum Ausdruck bringen und als unbedingte Ausdrücke (25) geschrieben werden,
- **Regeln,** die die zwischen den Objekten vorhandenen Beziehungen beschreiben und in der Form (24) dargestellt werden,
- **Anfragen** über Objekte und deren als Prädikat C formulierten Beziehungen

$$?(C) \tag{27}$$

In den folgenden Abschnitten wird gezeigt, wie das Beratungssystem in diese Form gebracht werden kann. Bildlich entspricht dies dem Übergang von der Konzeption des Beratungssystems im Bild 4 zur Realisierung eines solchen Beratungssystems als logisches Programm nach Bild 5.

[1)] Teil 1 erschien im Heft 5/1989, S. 204.

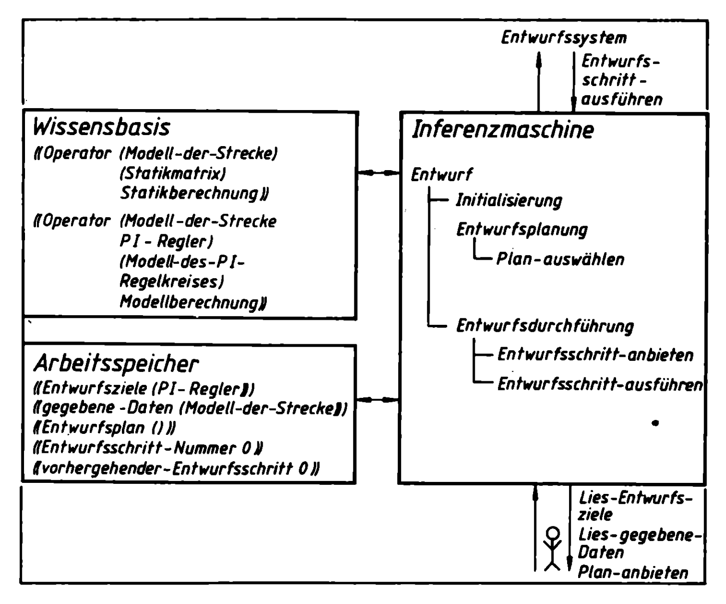


Bild 5. Implementierung des Beratungssystems in PROLOG

10. Die Wissensbasis des Beratungssystems

Die Wissensbasis des Beratungssystems enthält das Wissen über die Anwendungsbedingungen und Wirkungen der Moduln des Entwurfssystems (Abschn. 4., (W1) und Tafel 1). Dieses Wissen wird als unbedingte Ausdrücke (25) mit dem dreistelligen Prädikat „Operator" notiert. Die erste Stelle des Prädikats bezeichnet die Liste der Anwendungsvoraussetzungen, die zweite Stelle das Ergebnis (als einelementige Liste geschrieben) und die dritte Stelle den Modulnamen. So kann aus der zweiten Zeile von Tafel 1 der Ausdruck

((Operator (Modell-der-Strecke) (Statikmatrix) Statikberechnung))

gebildet werden. Da einige Moduln für mehrere Operationen eingesetzt werden können, beispielsweise der Stabilitätstest für die Strecke oder die Regelkreise mit P- oder PI-Regler, entstehen aus einer Zeile in Tafel 1 u. U. mehrere „Operator"-Klauseln. So entsprechen der ersten Zeile in Tafel 1 die Programmzeilen

((Operator (Modell-der-Strecke) (Stabilitaetsaussage-fuer-Strecke) Stabilitaetstest))
((Operator (Modell-des-P-Regelkreises) (Stabilitaetsaussage-fuer-P-Regelkreis) Stabilitaetstest))
((Operator (Modell-des-PI-Regelkreises) (Stabilitaetsaussage-fuer-PI-Regelkreis) Stabilitaetstest))

Unter Verwendung der Taxonomie (22) und den Erläuterungen aus Abschn. 2. entsteht auf diese Weise aus Tafel 1 die in Tafel 2 enthaltene Wissensbasis des Beratungssystems.

11. Der Arbeitsspeicher des Beratungssystems

Der Arbeitsspeicher entspricht Zeilen des logischen Programms, die die aktuellen Daten des Entwurfsprozesses enthalten. Für den im folgenden beschriebenen Ausbauzustand des Beratungssystems stellen diese Daten

- die Entwurfsziele (Zielzustand)
- die gegebenen Daten (Anfangszustand)
- den aktuellen Entwurfsplan
- die Nummer des zuletzt ausgeführten Entwurfsschrittes
- alle bereits ausgeführten Entwurfsschritte

dar (vgl. Abschn. 7.). Alte Daten können durch das Built-in-Prädikat KILL aus dem Speicher (d. h. aus dem PROLOG-Programm) gelöscht werden. Neue Daten werden mit ADDCL als unbedingte Ausdrücke (25) eingetragen.
Zu Beginn der Arbeit mit dem Beratungssystem wird der Arbeitsspeicher initialisiert. Dafür sind die alten Daten für Entwurfsziele und gegebene Datensätze zu löschen und die Entwurfsschritt-Nummer auf 1 zu setzen.

```
((Initialisierung)
  (KILL Entwurfsziele)
  (KILL gegebene-Daten)
  (KILL Entwurfsplan)
  (KILL Entwurfsschritt-Nummer)
  (ADDCL ((Entwurfsschritt-Nummer 1)) 0)
  (KILL Entwurfsschritt))                    (28)
```

Die Entwurfsaufgabe wird durch Einlesen der Liste der Entwurfsziele sowie der Liste der gegebenen Datensätze in den Arbeitsspeicher

Tafel 2. Wissensbasis des Beratungssystems

```
((Operator (Modell-der-Strecke)
           (Stabilitaetsaussage-fuer-Strecke)
                 Stabilitaetstest))
((Operator (Modell-des-PI-Regelkreises)
           (Stabilitaetsaussage-fuer-PI-Regelkreis)
                 Stabilitaetstest))
((Operator (Modell-des-P-Regelkreises)
           (Stabilitaetsaussage-fuer-P-Regelkreis)
                 Stabilitaetstest))
((Operator (Modell-der-Strecke)
           (Statikmatrix)
                 Statikberechnung))
((Operator (Statikmatrix)
           (Existenzaussage-fuer-PI-Regler)
                 Existenzpruefung))
((Operator (Modell-des-PI-Regelkreises Stabilitaetsaussage-fuer-PI-
           Regelkreis)
           (Uebergangsfunktion-des-PI-Regelkreises)
                 Simulation))
((Operator (Modell-des-P-Regelkreises Stabilitaetsaussage-fuer-P-
           Regelkreis)
           (Uebergangsfunktion-des-P-Regelkreises)
                 Simulation))
((Operator (Modell-der-Strecke P-Regler)
           (Modell-des-P-Regelkreises)
                 Modellberechnung))
((Operator (Modell-der-Strecke PI-Regler)
           Modell-des-PI-Regelkreises)
                 Modellberechnung))
((Operator (Uebergangsfunktion-des-PI-Regelkreises)
           (Bild-der-Uebergangsfunktion-des-PI-Regelkreises)
                 Grafik))
((Operator (Uebergangsfunktion-des-P-Regelkreises)
           (Bild-der-Uebergangsfunktion-des-P-Regelkreises)
                 Grafik))
((Operator (Modell-der-Strecke)
           (I-erweiterte-Strecke)
                 I-Erweiterung))
((Operator (Modell-der-Strecke)
           (Beobachter)
                 Beobachterentwurf))
((Operator (Modell-der-Strecke Beobachter)
           (P-Regler)
                 Polzuweisung-durch-Zustandsrueckfuehrung))
((Operator (I-erweiterte-Strecke Beobachter)
           (PI-Regler)
                 Polzuweisung-durch-Zustandsrueckfuehrung))
((Operator (Modell-der-Strecke Beobachter)
           (P-Regler)
                 Optimalreglerentwurf))
((Operator (I-erweiterte-Strecke Beobachter)
           (PI-Regler)
                 Optimalreglerentwurf))
((Operator (Modell-der-Strecke Stabilitaetsaussage-fuer-Strecke)
           (P-Regler)
                 Ausgangsrueckfuehrung))
((Operator (I-erweiterte-Strecke Existenzaussage-fuer-PI-Regler)
           (PI-Regler)
                 Ausgangsrueckfuehrung))
((Operator (Modell-der-Strecke Statikmatrix Existenzaussage-fuer-PI-
           Regler)
           (PI-Regler)
                 heuristische-Reglereinstellung))
```

eingetragen. Der Dialog mit dem Entwurfsingenieur erfolgt dabei mit den Built-in-Prädikaten R und PP bzw. P.

```
((Lies-Entwurfsziele)
  (P Entwurfsziele ?)
  (R X)
  (ADDCL ((Entwurfsziele X)) 0))                    (29)

((Lies-gegebene-Daten)
  (P gegebene-Daten ?)
  (R X)
  (ADDCL ((gegebene-Daten X)) 0))                   (30)
```

Bei der Anwendung dieser Prädikate werden die eingelesenen Listen zunächst an die Variable X gebunden und dann als Argumente unbedingter Ausdrücke in den Arbeitsspeicher geschrieben. Der mit dem Prädikat (29) eingelesene Zielzustand Z_1 aus (17) steht dann in der Form

```
((Entwurfsziele (Stabilitaetsaussage-fuer-P-Regelkreis
                 P-Regler) ))                        (31)
```

im Arbeitsspeicher.

12. Entwurfsplanung

Die Planung erfolgt durch Rückwärtsverkettung der durch die Prädikate „Operator" beschriebenen Moduln von den geforderten Entwurfszielen zu den gegebenen Daten (Abschn. 6.). In jedem Planungsschritt ist das (noch) zu lösende Planungsproblem durch folgende Informationen vollständig charakterisiert:

1. den bisher erstellten Teilplan Y
2. die Liste der noch zu erfüllenden Ziele Z.

Zur Vereinfachung des Planungsprozesses werden außerdem noch folgende Informationen mitgeführt:

3. die Liste der schon erfüllten Ziele x
4. die Liste der gegebenen Daten y

(Variablenbezeichnung wie in der zweiten Klausel von (32)). Unter den noch zu erfüllenden Zielen wird die Liste von allen denjenigen Datensätzen verstanden, die gebildet werden müssen, damit der bereits gebildete Teilplan anwendbar ist. Im nächsten Planungsschritt muß der Teilplan nach „vorn", d. h. in Richtung vom bereits bestimmten Teilplan zum Anfangszustand des Entwurfes, um eine Operator-Anwendung verlängert werden, wobei ein Ziel aus der Liste Z durch diesen Operator erfüllt werden soll.
Jeder Planungsschritt setzt sich also aus der Bestimmung eines anwendbaren Operators z, der eines der noch zu erfüllenden Ziele Z erfüllt, der Bildung der neuen Listen X1 und Y1 schon erfüllter bzw. noch zu erfüllender Ziele sowie der Verlängerung des alten Teilplanes um einen Schritt in (z | Y) zusammen. Das Planungsproblem ist gelöst, wenn die Liste der noch zu erfüllenden Teilziele leer ist, wobei der in der zweiten Stelle des Prädikates „Plan" zusammengestellte Teilplan auf das erste Argument übertragen wird.

```
((Plan X X ( ) Y Z))
                  Plan * Teilplan * zu erfüllende Ziele * schon
                  erfüllte Ziele * gegebene Daten
((Plan X Y Z x y)
  (anwendbarer-Operator z Z x Y)
  (erfuellte-Ziele X1 Z z)
  (neue-Ziele Y1 Z z y)
  (Plan X (z | Y) Y1 X1 y))                          (32)
```

Welcher Operator (Operator X Y Z) aus der Wissensbasis „anwendbarer-Operator" (X Y Z) ist, wird durch folgende Bedingungen fixiert (Variablenbezeichnung wie in (33)):

- Der Operator soll ein Ziel der Liste x erfüllen.
- In den Anwendungsbedingungen X des Operators soll kein Datensatz auftreten, der zur Liste der bereits erfüllten Ziele y gehört, denn dies hieße, daß dieser Datensatz in dem noch zu bestimmenden Teil des Planes ein zweites Mal gebildet werden müßte.
- Der Operator soll nicht zum bereits gebildeten Teilplan z gehören, denn dann müßte er mehr als einmal angewendet werden (z. B. zweimalige Berechnung einer Ausgangsrückführung).

```
          Operator*Ziele*erfüllte Ziele*verwendete Operatoren
((anwendbarer-Operator (X Y Z) x y z)
  (Operator X Y Z)
  (haben-gemeinsames-Element Y x)
  (NOT haben-gemeinsames-Element X y)
  (NOT Element-der-Menge (X Y Z) z))                 (33)
```

Die Liste der schon erfüllten Ziele entsteht durch Anhängen des im Ergebnisteil des Operators (Y (Z) x) stehenden Ziels Z an die bisherige Liste X.

```
                        neue Liste * alte Liste * Operator
((erfuellte-Ziele (X | Y) Y (Z (X) x)))              (34)
```

Die Liste der neuen Ziele ergibt sich aus der Liste der bisher erfüllten Ziele unter Wegnahme des durch den Operator erfüllten Zieles und Hinzufügen aller zu den Anwendungsbedingungen des Operators gehörenden Ziele, die nicht gegebene Daten sind. Dabei soll jedes Ziel nur einfach auftreten.

```
          neue Liste * alte Liste * Operator * gegebene Daten
((neue-Ziele X Y (Z x y) z)
  (ISALL X1 Y1 (Element-der-Menge Y1 Y)
               (NOT Element-der-Menge Y1 x))
  (ISALL X Z1  (OR ((Element-der-Menge Z1 X1))
                   ((Element-der-Menge Z1 Z)
                    (NOT Element-der-Menge Z1 X1)
                    (NOT Element-der-Menge Z1 z)))))) (35)
```

Vorausgesetzt, daß die Entwurfsziele und die gegebenen Daten bereits im Arbeitsspeicher eingetragen sind, besteht die Planung in der Suche nach einer Lösung Z, die das Prädikat „Plan" erfüllt.

```
((Entwurfsplanung1)
  (Entwurfsziele X)
  (gegebene-Daten Y)
  (Plan Z ( ) X ( ) Y)
  (ADDCL ((Entwurfsplan Z)) 0))                      (36)
```

Nach dem Aufruf

```
?((Entwurfsplanung1))
```

wird der PROLOG-Interpreter also nach einer Klausel für das Prädikat „Entwurfsziele" suchen, die Eintragung (31) finden und die Liste (Stabilitaetsaussage-fuer-PI-Regelkreis PI-Regler) an X binden, womit das erste Teilziel der Klausel (36) erfüllt ist. In gleicher Weise wird die Liste der gegebenen Daten an Y gebunden. Das Prädikat „Plan" wird dann mit leeren Listen für den bereits erstellten Teilplan und die bereits erfüllten Ziele und gebundenen Listen X und Y aufgerufen. In (32) bis (36) sind die Eigenschaften eines einzelnen Planungsschrittes *beschrieben.* Wie eine Liste von Operatoren, die als Liste Z das Prädikat „Plan" erfüllt, gesucht werden kann, wird vom Interpreter selbständig organisiert.

13. Entwurfsdurchführung

Bei der Entwurfsdurchführung hat das Beratungssystem den jeweils im aktuellen Entwurfsplan als erstes stehenden Operator zur Anwendung zu bringen. Ein sofortiger Aufruf des entsprechenden Moduls des Entwurfssystems ist möglich (vollständige Automatisierung des Entwurfes), aber i. allg. nicht zweckmäßig, weil die schrittweise Realisierung des aufgestellten Planes in Abhängigkeit von den erreichten Entwurfsergebnissen möglicherweise durch den Entwurfsingenieur abgebrochen und der Plan modifiziert werden soll. Der Entwurfsschritt wird deshalb zuerst dem Nutzer angeboten und erst nach dessen Bestätigung ausgeführt.
Nach Ausführung des Entwurfsschrittes ist der Wert für die Entwurfsschritt-Nummer zu aktualisieren, der Entwurfsschritt in den Arbeitsspeicher einzutragen und der aktuelle Entwurfsplan um den verwendeten Operator zu verkürzen. Der Entwurf ist beendet, wenn der Entwurfsplan leer ist.

```
0 ————————————→ 1 ————————————→ 2 ————————————→ 3 ...→
((Entwurfsschritt 1 0                ((Entwurfsschritt 2 1                ((Entwurfsschritt 3 2
(Modell-der-Strecke)                 (Modell-der-Strecke                  (Modell-der-Strecke
(Stabilitätsaussage-für-Strecke)     Stabilitätsaussage-für-Strecke)      P-Regler)
Stabilitätstest))                    (P-Regler)                           (Model-des-P-Regelkreises)
                                     Ausgangsrückführung))                Modellberechnung))
```

Bild 6. Entwurfspfad in PROLOG-Notierung

```
((Entwurfsdurchfuehrung1)
 (Entwurfsplan ( ))
 (PP) (PP Plan ist vollstaendig ausgefuehrt)
 (PP Ende des Entwurfes))
((Entwurfsdurchfuehrung1)
 (Entwurfsplan (X ¦ Y))
 (Entwurfsschritt-Nummer Z)
 (Entwurfsschritt-anbieten X Z x)
 (IF (EQ x n) ((PP Mit neuer Entwurfsaufgabe fortsetzen))
         (( / )))
 (Entwurfsschritt-ausfuehren X Z)
 (KILL Entwurfsplan)
 (ADDCL ((Entwurfsplan Y)) 0)
 (Entwurfsdurchfuehrung1))                                (37)
              Operator * Entwurfsschritt-Nr. * Antwort
((Entwurfsschritt-anbieten (X Y Z) x y)
 (PP x-ter Entwurfsschritt: Z)
 (P ausfuehren ?)
 (R z)
 (EQ y z))                                                (38)
                       Operator * Entwurfsschritt-Nr.
((Entwurfsschritt-ausfuehren X Y)
 (SUM x 1 Y)
 (ADDCL ((Entwurfsschritt Y Z X)) Z)
 (KILL Entwurfsschritt-Nummer)
 (SUM Y 1 x)
 (ADDCL ((Entwurfsschritt-Nummer x)) 0))                 (39)
```

Die durchgeführten Entwurfsschritte werden als unbedingte Ausdrücke für das Prädikat „Entwurfsschritt" in das Programm eingetragen, wobei die ersten beiden Stellen die Nummer des Entwurfsschrittes und des vorangegangenen Entwurfsschrittes enthalten und die dritte Stelle den Operator beschreibt. Der Entwurfsweg aus Bild 3 ist deshalb in der im Bild 6 gezeigten Form im Arbeitsspeicher enthalten. Die den Zustand charakterisierenden Daten sind in „Entwurfsschritt" enthalten und die Zustände selbst in der Reihenfolge ihrer Erzeugung bei der Entwurfsdurchführung durchnumeriert.

14. Ein einfaches Beratungssystem

Unter Verwendung der in den vorangegangenen Abschnitten entwickelten Teilschritte für die Entwurfsplanung und die Entwurfsdurchführung kann ein einfaches Beratungssystem aufgebaut werden, das die im Bild 4 konzipierte Struktur hat (Bild 5). Die Wissensbasis enthält die als Prädikat „Operator" beschriebenen Anwendungsvorschriften der Moduln des Entwurfssystems. Im Arbeitsspeicher stehen die den Zustand des Entwurfs vollständig beschreibenden Informationen (Abschn. 11.). Die Inferenzmaschine umfaßt Prädikate für die Initialisierung des Arbeitsspeichers, die Entwurfsplanung und die Entwurfsdurchführung sowie die für die Definition dieser Prädikate notwendigen weiteren Prädikate. Diese Prädikate sind ausreichend, um folgende Schritte unter Benutzung des in der Wissensbasis stehenden Wissens über die Anwendungsmöglichkeiten des Entwurfssystems ausführen zu können.

(1) Einlesen der Entwurfsaufgabe
(2) Bestimmung eines Entwurfsplanes
(3) Steuerung des Entwurfssystems zur Ausführung von Entwurfsschritten entsprechend dem Entwurfsplan.

Dieser Ablauf wird durch die Anwendung des Prädikates „Entwurf" erreicht.

```
((Entwurf)
 (PP) (PP ***** BERATUNGSSYSTEM ******) (PP)
 (Initialisierung)
 (Entwurfsplanung) ( / )
 (Entwurfsdurchfuehrung))                                 (40)
```

Das Beratungssystem ist vollständig durch die Klauseln (26) bis (40), Tafel 2 und Anhang 1 beschrieben. Die geringfügige Erweiterung, die im Anhang 1 angegeben ist, ermöglicht es, im Schritt (2) mehrere Entwurfspläne nacheinander zu ermitteln, dem Entwerfer anzubieten und schließlich den ausgewählten Entwurfsplan als gültigen Plan in den Arbeitsspeicher einzutragen. Einen Eindruck von der Arbeit mit diesem einfachen Beratungssystem vermittelt die Wiedergabe des Dialogs zwischen Beratungssystem und Entwurfsingenieur im Anhang 2.

15. Softwaretechnologische Merkmale des Beratungssystems

Aufgrund seiner Struktur und der Implementierung in PROLOG weist das Beratungssystem typische Merkmale von Expertensystemen auf (vgl. [9]):

- Wissensbasis und Wissensverarbeitung sind in getrennten Komponenten zusammengefaßt. Das hier für den rechnergestützten Entwurf linearer Regelungen entwickelte Beratungssystem kann deshalb ohne Probleme auch im Zusammenspiel mit andersartigen Entwurfssystemen eingesetzt werden, die die im Abschn. 3. zusammengestellten Charakteristika aufweisen. Dafür muß lediglich die Wissensbasis, d. h. die Definition des Prädikats „Operator", ausgetauscht werden.
- Das Beratungssystem ist deskriptiv programmiert. Für jedes Prädikat wurde lediglich eine Beschreibung bzw. Definition angegeben. Wann auf diese Prädikate zurückgegriffen wird und wie Lösungen gefunden werden, die diese Prädikate erfüllen, wird vom PROLOG-Interpreter selbständig organisiert. Die Planungskomponente ist ein offensichtliches Beispiel dafür, daß beim Programmieren lediglich die in *einem* Planungsschritt wichtigen Merkmale beschrieben werden müssen und der Interpreter die für die gesamte Planung notwendigen Schritte aufgrund dieser Beschreibung selbst findet.
- Veränderungen und Erweiterungen des Beratungssystems sind nahezu problemlos möglich. Es müssen lediglich die Definitionen der Prädikate verändert oder erweitert werden. Solange sich in den Argumenten des Prädikates nichts ändert, bedeutet eine solche Änderung nur den Austausch der Definitionen.

Die Zweckmäßigkeit, mit der KI-Sprachen für die Prototypentwicklung von Beratungs- und Expertensystemen eingesetzt werden können, kann sich der Leser unschwer durch die Überlegung vergegenwärtigen, welchen Aufwand es gekostet hätte, das hier beschriebene Beratungssystem in einer algorithmisch orientierten Sprache (PASCAL, FORTRAN u. a.) zu schreiben.

16. Erweiterungsmöglichkeiten und Ausblick

Ziel des Beitrags war es, die Aufgaben eines Beratungssystems für den rechnergestützten Entwurf zusammenzustellen, eine durch Prinzipien der Wissensverarbeitung begründete Konzeption anzugeben und die Realisierung des Beratungssystems als logisches Programm im Detail zu erläutern. Dabei konnte aus Platzgründen nur auf eine einfache Version eines solchen Beratungssystems eingegangen werden. Unter Beibehaltung des vorgestellten Konzeptes sind jedoch Erweiterungen möglich, auf die im folgenden hingewiesen werden soll, um gleichzeitig die Wirkungsbreite eines solchen Beratungssystems aufzuzeigen.

Entwurfsplanung

— Da der aktuelle Entwurfszustand vollständig im Arbeitsspeicher enthalten ist, kann der Entwurf zu einem beliebigen Zeitpunkt unterbrochen und später an der alten Stelle fortgesetzt werden. Dafür müssen im Prädikat „Entwurf" in (40) Vorkehrungen getroffen werden, um die Initialisierung zu umgehen und den noch nicht realisierten Entwurfsplan für die Weiterbearbeitung der alten Aufgabe anzubieten.
— Wenn es der Entwurfsingenieur nach der Ausführung einiger Entwurfsschritte als unzweckmäßig erachtet, weiterhin entsprechend dem zunächst akzeptierten Entwurfsplan vorzugehen, kann die Entwurfsdurchführung abgebrochen und in die Planungsphase zurückgekehrt werden. Bei der Neuplanung muß berücksichtigt werden, daß außer den gegebenen Daten noch die auf dem Pfad vom Startzustand zum aktuellen Zustand erzeugten Daten als „Fakten" zur Verfügung stehen.
— Die Planung kann durch heuristische Regeln so gelenkt werden, daß „günstige" Pläne vorrangig erzeugt werden. Dafür ist das Prädikat „anwendbarer-Operator" gegenüber (33) zu verändern. Es ist erst die Kompromißmenge aller anwendbaren Operatoren (nach den Bedingungen aus Abschn. 12.) zu bilden und aus dieser der entsprechend heuristischen Regeln günstigste Operator auszuwählen. Heuristische Regeln können z. B. sein:
  * Nimm einen Operator, dessen Anwendungsbedingungen ausschließlich gegebene Daten enthalten (und der somit keine neuen Teilziele erzeugt, vgl. Abschn. 6.)!
  * Nimm einen Operator, dessen Anwendungsbedingungen mindestens einen gegebenen Datensatz enthalten!
  * Nimm einen Operator, dessen Anwendungsbedingungen nur Datensätze enthalten, die bereits in der Liste der zu erfüllenden Ziele eingetragen sind!
— Es kann dem Entwerfer freigestellt werden, selbst Entwurfspläne vorzuschlagen. Dann ist das „inverse Planungsproblem" zu lösen, d. h., es ist mit Hilfe der Wissensbasis zu überprüfen, ob die angegebene Folge von Operatoren mit den vorhandenen Moduln des Entwurfssystems realisierbar ist.

Entwurfsdurchführung

— Die Aufrufe der Moduln können vollständig zusammengestellt werden. Dafür sind auch die zu verarbeitenden bzw. der zu erzeugende Datensatz zu benennen. Beide Informationen können vollständig aus dem Arbeitsspeicher-Inhalt herausgesucht werden.
— Der vom Entwerfer akzeptierte Entwurfsplan kann ohne weitere Nutzereingriffe vom Beratungssystem realisiert werden, wenn darauf verzichtet wird, dem Entwurfsingenieur jeden Schritt zur Bestätigung anzubieten (vollständige Automatisierung des Entwurfes).

Dokumentation des Entwurfsprozesses

— Das Beratungssystem kann die Vergabe zweckmäßiger Namen für die durch die Entwurfsmoduln erzeugten Dateien übernehmen und diese Namen zusammen mit der Semantik der Datei im Arbeitsspeicher festhalten, z. B. in der Form

```
((Modell-der-Strecke MdS.DAT))
  (Semantik)        (Filename)
```

— Vom Beratungssystem kann jederzeit Auskunft über den Entwurfsplan, den durchlaufenen Entwurfsweg, die dabei erzeugten Dateien und den aktuellen Zustand des Entwurfes gegeben werden.

Das PROLOG-Programm eines Beratungssystems, das diese erweiterten Funktionen übernimmt, kann Interessenten vom Autor zur Verfügung gestellt werden.
Durch Erweiterungen der Konzeption bzw. der vorgestellten Implementierung kann das Beratungssystem auch dafür eingesetzt werden,

— um Forderungen an die Gestaltung des Planes zu erfüllen (z. B. sollen besonders zweckmäßige Folgen von Moduln zuerst erzeugt bzw. unzweckmäßige oder verbotene Folgen ausgeschlossen werden)
— um die bei der Abarbeitung von Moduln des Entwurfssystems möglicherweise auftretenden Interaktionen mit dem Entwurfsingenieur auf den der Entwurfsaufgabe angemessenen Teil zu beschränken
— um die bei der Entwurfsdurchführung entstehenden Informationen über das Entwurfsproblem (z. B. der *Inhalt* der Stabilitätsaussage für die Strecke) für die rechtzeitige Veränderung des Entwurfsplanes zu nutzen.

Die letzte Erweiterung deutet an, daß bei rein strukturellen Betrachtungen als Grundlage des Beratungssystems wenig Möglichkeiten bestehen, Erfahrungen des Entwerfers mit bestimmten, durch eine charakteristische Art des dynamischen Verhaltens ausgezeichneten Klassen von Regelstrecken in die Wissensbasis aufzunehmen. Da sich diese, zusätzlich zu Tafel 2 aufzunehmenden, Wissenselemente auf bestimmte Parameterkombinationen des Streckenmodells beziehen, muß der Dialog zwischen Beratungssystem und Entwurfssystem so erweitert werden, daß im Arbeitsspeicher Fakten eingetragen werden können, die das Vorhandensein derartiger Parameter beschreiben. Außerdem muß untersucht werden, welche Entwerfer-Erfahrungen verallgemeinerbar sind und in welcher Form sie in die Wissensbasis aufgenommen werden können. Eine diesbezügliche Studie zu Frequenzbereichsverfahren [12] zeigt, daß für eine derartige Erweiterung des Beratungssystems eine gründliche Aufarbeitung etablierter Entwurfsverfahren notwendig ist.

Anhang 1. Vervollständigung des PROLOG-Programms

```
((Entwurfsplanung)
  (PP) (PP *** Entwurfsplanung ***) (PP)
  (Lies-Entwurfsziele)
  (Lies-gegebene-Daten)
  (Plan-auswaehlen X)
  (KILL-Entwurfsplan)
  (ADDCL ((Entwurfsplan X)) 0))
((Plan-auswaehlen X)
  (Entwurfsziele Y)
  (gegebene-Daten Z)
  (Plan1 X ( ) Y ( ) Z))
((Plan1 X X ( ) Y Z)
  (Plan-anbieten X x)
  (IF (EQ x j) (( / )) ((FAIL))) )
((Plan-anbieten X Y)             Plan * Antwort
  (PP) (PP Plan:) (Print-Plan X 1)
  (P Plan ok ?)
  (R Z)
  (EQ Y Z))
((Print-Plan ( ) X))
((Print-Plan ((X Y Z) | x) y)       Plan * Antwort
  (PP) (PP y-ter Schritt: Z)
  (PP Eingaben: X)
  (PP Ergebnis: Y)
  (SUM y 1 z)
  (Print-Plan x z))
((Entwurfsdurchfuehrung)
  (PP) (PP *** Entwurfsdurchfuehrung ***) (PP)
  (Entwurfsdurchfuehrung1))
```

Anhang 2. Ausschnitt aus dem Dialog Beratungssystem — Nutzer

(Nutzereingaben sind halbfett gesetzt)

?((Entwurf))

***** BERATUNGSSYSTEM *****
*** Entwurfsplanung ***
Entwurfsziele ? **(Bild-der-Uebergangsfunktion-des-PI-Regelkreises)**
gegebene Daten ? **(Modell-der-Strecke)**
Plan:
1-ter Schritt Beobachterentwurf
Eingaben: (Modell-der-Strecke)
Ergebnis: (Beobachter)

2-ter Schritt I-Erweiterung
Eingaben: (Modell-der-Strecke)
Ergebnis: (I-erweiterte-Strecke)
3-ter Schritt Polzuweisung-durch-Zustandsrueckfuehrung
Eingaben: (I-erweiterte-Strecke Beobachter)
Ergebnis: (PI-Regler)
4-ter Schritt Modellberechnung
Eingaben: (Modell-der-Strecke PI-Regler)
Ergebnis: (Modell-des-PI-Regelkreises)
5-ter Schritt Stabilitaetstest
Eingaben: (Modell-des-PI-Regelkreises)
Ergebnis: (Stabilitaetsaussage-fuer-PI-Regelkreis)
6-ter Schritt Simulation
Eingaben: (Modell-des-PI-Regelkreises Stabilitaetsaussage-fuer-PI-Regelkreis)
Ergebnis: (Uebergangsfunktion-des-PI-Regelkreises)
7-ter Schritt Grafik
Eingaben: (Uebergangsfunktion-des-PI-Regelkreises)
Ergebnis: (Bild-der-Uebergangsfunktion-des-PI-Regelkreises)
Plan ok ? j
*** Entwurfsdurchfuehrung ***
1-ter Entwurfsschritt Beobachterentwurf
ausfuehren ? j
2-ter Entwurfsschritt I-Erweiterung
ausfuehren ? j

Literatur

[1] Barr, A.; Feigenbaum, E. H.: The Handbook of Artificial Intelligence. Los Altos: William Kaufman 1982.

[2] Clark, K. L.; McCabe, F. G.: micro-PROLOG: Programming in Logic. Englewood Cliffs: Prentice-Hall 1984.

[3] Clocksin, W. F.; Mellish, C. S.: Programming in Prolog. Berlin: Springer-Verlag 1981.

[4] Hans, U.; Petersohn, U.; Pitschke, J.: Programmieren in micro-PROLOG. Lehrhefte „Ausbildung Informatik", TU Dresden 1987.

[5] IMGRE-1. Nutzerhandbuch. ZKI der AdW, Dresden 1986.

[6] Korn, U.; Wilfert, H. H.: Mehrgrößenregelungen. Berlin: VEB Verlag Technik 1982.

[7] Lewin, D. R.; et al.: ROBEX — robust control synthesis via expert system. IFAC-Congress, Munich 1987, pp. 369—374.

[8] Lunze, J.: Determination of robust multivariable I-controllers by means of experiments and simulation. Syst. Anal. Model. Simul. 2 (1985) 4, pp. 227—249.

[9] Lunze, J.: Wissensbasierte Systeme. msr, Berlin 30 (1987) 10, S. 437—443, und 11, S. 502—508.

[10] Lunze, J.: Strukturelle Analyse rückgekoppelter Systeme mit Hilfe von PROLOG. msr, Berlin 31 (1988) 9, S. 386—390.

[11] Nolan, P. J.: An intelligent assistant for control system design. 1st Conf. on Applications of AI in engineering problems, Southampton 1986.

[12] Pang, G. K. H.; MacFarlane, A. G. J.: An expert systems approach to computer-aided design of multivariable systems. Berlin: Springer-Verlag 1987.

[13] Pötschke, D.; Lunze, J.: Künstliche Intelligenz und Automatisierungstechnik. Band 237 der REIHE AUTOMATISIERUNGSTECHNIK. Berlin: VEB Verlag Technik 1989.

[14] Taylor, J. H.; Frederick, D. K.; James, J. R.: An expert system scenario for computer-aided control engineering. American Control Conf., San Diego 1984, pp. 120—128.

[15] Taylor, J. H.; et al.: Expert-aided control engineering environment for nonlinear systems. IFAC-Congress, Munich 1987, vol. 6, pp. 363 to 374.

[16] Trankle, R. L.; Sheu, P.; Rabin, U. H.: Expert system architecture for control system design. American Control Conf., Seattle 1986, pp. 1163—1169.

msr 8956

L. Rothe

Optimierungsalgorithmen der Aperturbelegung höchstfrequenter Strahlungssensoren

0. Einleitung

Sensoren sind als Funktionskomponenten zu betrachten, deren Konzipierung ausgeprägter als in früheren Etappen der Automatisierungstechnik an die Spezifik der Einsatzfälle gebunden ist. Charakteristisch für sensorische Systeme bzw. Komponenten ist jedoch auch die zunehmende Notwendigkeit funktioneller Integration im Prozeß der Sensorkonzeption. Mit dem primären Vermögen, aktuelle Prozeß- und Zustandsparameter selektiv erfassen zu können, sind lediglich unbewertete Grundfunktionen automatisierungsbedingter Erkennungssysteme realisierbar. Derartige Grundfunktionen existieren jedoch statisch und verleihen funktionell bedingt keinerlei Adaptivität. Die Implementierung adaptiver Eigenschaften bzw. adaptiven Vermögens ist in diesem Zusammenhang in sehr differenzierter Weise an die Voraussetzung künstlicher Intelligenz gebunden. In den Fällen, in denen diese erforderlich wird, ist jedoch die Systembetrachtung zugrunde zu legen, d. h., der Sensor und intelligente Komponenten sind als einander bedingende Funktionskomponenten aufzufassen bzw. zu konzipieren. Adaptive Mechanismen bilden die Lösungsgrundlage spezifischer Steuerungsaufgaben bzw. -probleme.

Der Beitrag hat das Ziel, die Adaptivitätsproblematik oder auch das Vermögen der zielgerichteten und ergebnisorientierten Anpaßbarkeit vorgegebener Parameter und Zustände eines Systems an die Parameter bzw. Eigenschaften des Zielsystems auf die Sensorproblematik zu übertragen. Transparent wird dies insbesondere für den Fall der Strahlungssensoren in Form der praxisrelevanten Sensorarrayanordnungen.

Wenn auch der Diskussionsschwerpunkt wegen der Deutlichkeit der physikalischen Phänomene auf den Spektralbereich des hochfrequenten Radiowellenspektrums beschränkt wird, ist damit die Übertragbarkeit der Vorgehensweise auf die sich anschließenden Spektralbereiche der kurzwelligeren elektromagnetischen Strahlung keinen grundsätzlichen Beschränkungen unterlegen.

Dr. sc. techn. Lutz Rothe (37) studierte von 1969 bis 1973 an der Sektion Informationstechnik und theoretische Elektrotechnik der TH Ilmenau. Anschließend dort wissenschaftlicher Assistent. 1978 Promotion A (Mikrowellentechnik). Von 1980 bis 1986 Praxistätigkeit. Danach wissenschaftlicher Oberassistent am Wissenschaftsbereich Informationstechnik der TH Leipzig. 1988 Promotion B.

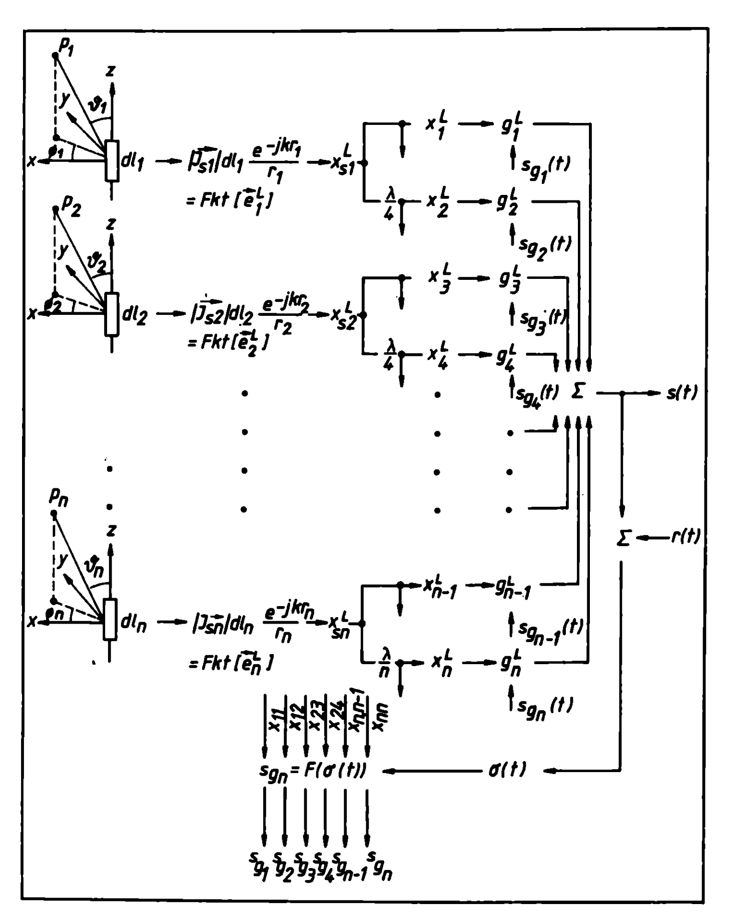


Bild 1. Modellierungskonfiguration der Anpassungskonzeption von Sensor- und Feldcharakteristik

$x_i(t) - jx_{i+1}(t) \rightarrow g_i(t) + jg_{i+1}(t) \rightarrow g_{s_i}^L(t) x_{s_i}(t)$

$S_{g_i}(t), g_{g_{i+1}}(t)$

Bild 2. Summierungsvorschrift der Elementsignale des Sensorsystems

1. Optimierungsprozeß

Mit der Zuordnung einer allgemeinen Erkennungsaufgabe läßt sich das Optimierungsproblem als spezifische Modifikation eines Erkennungsvorganges formulieren. Unter dieser Voraussetzung ist damit die Problematik als Erkennung von Wertemustern mit der Eigenschaft der partiellen Überdeckung durch Fremd- bzw. Störmuster sowie weitergehend als Erfassung von gestörten oder ungestörten Wertemustern mit der Eigenschaft der stochastischen Veränderlichkeit der Zustandsfunktionale beschreibbar.

Die Erkennung von Wertemustern beinhaltet in diesem Zusammenhang sowohl das Problem der Signal- bzw. Informationserfassung als auch das Problem der Signalverarbeitung und -analyse. Grenzt man das Sensorproblem über die funktionelle Basis in Form der Kopplungsbedingungen zwischen sensorisch zu erfassender und physikalischer Größe ein, wird die Korrelation zwischen aufgabenbedingter räumlicher Distribution der Signalerfassung und der Meßwerterfassung selektiver physikalischer Informationen voraussetzungsgemäß systembestimmend.

Postuliert man, daß reale Feldbedingungen mit der Ausbildung von Interferenzfeldern infolge der Überlagerung des direkten Strahlungsfeldes mit reflexions- sowie brechungsbedingten indirekten Feldern oder kohärenten Störstrahlungsanteilen bzw. ausschließlich der Überlagerung indirekter Felder verbunden sind, ist der Optimierungsprozeß auf die amplituden- und phasenmäßige Zustandskonfiguration der Aperturbelegung des Strahlungssensors und damit auf die Synthese der Amplituden- und Phasenfunktion gemäß des Optimierungszieles zu beziehen. Legt man der Anpassungskonzeption von Sensor- und Feldcharakteristik des realen Raumes die Modellierungskonfiguration gemäß Bild 1 zugrunde, läßt sich über die Einführung eines Referenzsignals eine Steuerfunktion der komplexen Signalbewertung eines n-elementigen Sensorsystems ableiten.

Voraussetzungsgemäß wird ein n-elementiges Sensorsystem betrachtet, dessen komplex bewertete Elementsignale summierbar sind:

$$\sum_{i=1}^{n} g_{s_i}^L x_{s_i}^L(t) = \sum_{i=1}^{n} g_{s_i}^L \left|x_{s_i}^L\right| e^{j(\omega t+\psi_i)} \tag{1}$$

mit

$$x_{s_i}^L(t) = \left|x_{s_i}^L\right| e^{j(\omega t+\psi_i)} \tag{2}$$

bzw.

$$x_{s_i}^L(t)\sqrt{2} = x_i(t) - jx_{i+1}(t) \tag{3}$$

$$g_{s_i}^L(t) = g_i(t) + jg_{i+1}(t)\,. \tag{4}$$

Gemäß der Summierungsvorschrift nach Bild 2 gilt:

$$\sum_{i=1}^{n} g_{s_i}^L(t)\, x_{s_i}^L(t) = \sum_{i=1}^{2n} g_i^L(t)\, x_i^L(t). \tag{5}$$

Wählt man als Referenzsignal ein Elementsignal

$$r^2(t) = n_{g_k}^L x_k^L(t) = n_{g_k}^L x_k\, e^{j(\omega t+\psi_k)} \tag{6}$$

unter der Voraussetzung

$$g_k^L x_k^L(t) = g_k^L x_k\, e^{j(\omega t+\psi_k)} = y^L(t) \tag{7}$$

mit

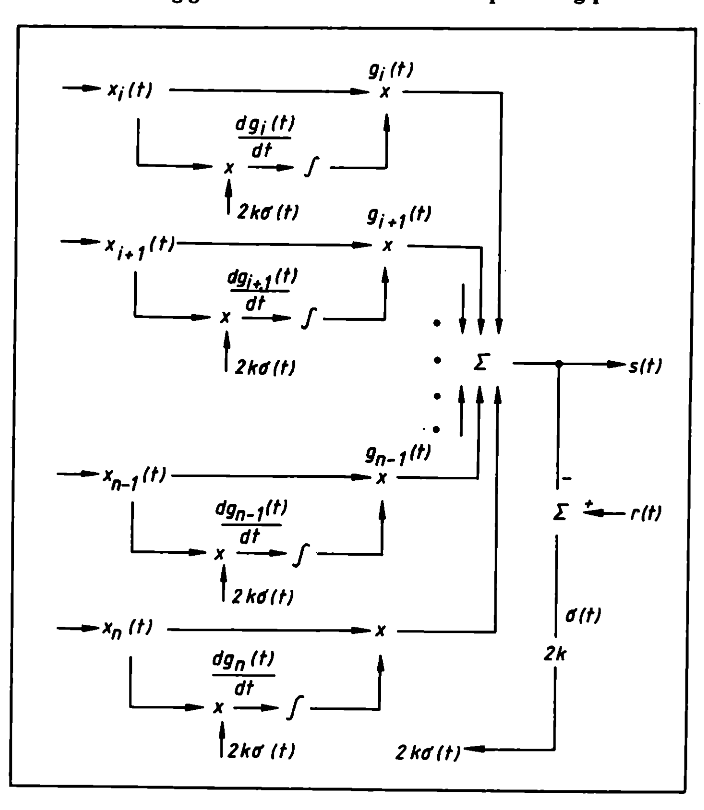
$$g_k^L = 1\,, \tag{8.1}$$

$$x_k^L(t) = x_{N+1}^L(t)\,, \tag{8.2}$$

$$\sum_{i=1}^{n} x_i^L(t) + x_k^L(t) = y^L(t)\,, \tag{8.3}$$

$$\sum_{i=1}^{n} x_i^L(t) = 0 \tag{8.4}$$

Bild 3. Zeitabhängigkeit der Steuerfunktionen des Optimierungsprozesses



oder

$$g_k^L = 1\,, \tag{9.1}$$

$$x_k^L(t) \in x_i^L(t) \tag{9.2}$$

$$\sum_{i=1}^{k-1} x_i^L(t) + \sum_{k+1}^{n} x_i^L(t) = 0\,, \tag{9.3}$$

läßt sich die Steuerfunktion der komplexen Signalbewertung des n-elementigen Sensorsystems ableiten:

$$\sigma^L(t) = r^L(t) - \sum_{i=1}^{2n} g_i^L x_i^L(t) \tag{10}$$

$\sum_{i=1}^{2n} g_i^L x_i^L(t)$ Summensignal des n-elementigen Sensorsystems;

$r^L(t)$ Referenzsignal; $s(t) = f[\sigma(t)]$ Steuerfunktion

bzw.

$$\overline{\sigma^2(t)} = \overline{r^2(t)} - 2\sum_{i}^{2n} g_i \overline{r(t)\,x_i(t)} + \sum_{i=1}^{2n}\sum_{j=1}^{2n} g_i g_j \overline{x_i'(t)\;x_j'(t)}\,. \tag{11}$$

Die Auswertung des Extremwertproblems

$$\operatorname{grad}_g \overline{(\sigma^2(t))} = 0 \tag{12}$$

ermöglicht die Berechnung der optimierten Werte der Funktionen $g(t)$:

$$\operatorname{grad}_g[\overline{\sigma^2(t)}] = -2\sum_{i=1}^{2n} \overline{r(t)\;x_i(t)} + 2\sum_{i=1}^{2n} g\overline{x_i(t)\,x_j(t)} \tag{13.1}$$

und

$$g_{\mathrm{opt.}} = \sum_{i=1}^{2n} \overline{r(t)\,x_i(t)}\left[\sum_{i=1}^{2n} \overline{x_i(t)\,x_j(t)}\right]^{-1} \tag{13.2}$$

Die Funktion $g_{\mathrm{opt.}}$ ist dabei die Zielfunktion des Optimierungsprozesses als zeitabhängiger Vorgang, dessen zeitliche Änderungsvorschrift festzulegen ist.

Auszugehen wäre in diesem Fall von der Zuordnung der zeitlichen Änderung der Steuerfunktion g in der Form

$$\lim_{\Delta t \to 0} \frac{\Delta g_i}{\Delta t} = \frac{\mathrm{d}g_i}{\mathrm{d}t} = \mathrm{F}(\operatorname{grad}_{g_i}[\overline{\sigma^2(t)}])\,. \tag{14}$$

Mit

$$\operatorname{grad}_{g_i}[\overline{\sigma^2(t)}] = -2x_i(t)\,r(t) + 2\sum_{j=1}^{2n} \overline{x_i(t)\;x_j(t)}\;g_i$$

läßt sich der zeitliche Prozeß des Optimierungsalgorithmus formulieren (Bild 3)

$$g_i(t) = g_i(0) + 2C\int_0^t x_i(t)\;\sigma(t)\;\mathrm{d}t\,. \tag{15}$$

Für den Fall der Elevationswinkel oberhalb des $\sqrt{2}$-fachen Wertes des Hauptstrahlsektors des Strahlungsfeldsensors erfolgt die Detektion interferierender Signale über die Nebenstrahlungsmaxima. In diesem Fall wäre die Monopulsmethode anwendbar.

Für den Fall der Elevationswinkel unterhalb des $\sqrt{2}$-fachen Wertes des Hauptstrahlsektors über dem Horizont besteht aufgrund der Einwirkung interferierender Signale in den Hauptstrahlungsbereich das Problem der Störsignal- bzw. Interferenzselektion.

2. Optimierungsverfahren

Bekannte Verfahren basieren auf der Signalbewertung durch räumliche Bandpaßfilterung mit einer Sperrcharakteristik für alle Einfallsrichtungen unterhalb des Horizonts [1].

Weitere Verfahren bestimmen das räumliche Leistungsspektrum durch die Anwendung von Methoden maximaler Entropie [2] bis [5].

Voraussetzung dieser Lösungsmöglichkeit bildet jedoch die Erfüllung der Dekorrelationsbedingung des Primärsignals und seiner Mehrwegekonfiguration.

Der Optimierungsalgorithmus des Detektionsproblems ist über die Modellierung des Überlagerungsfalles formulierbar (Bild 4).

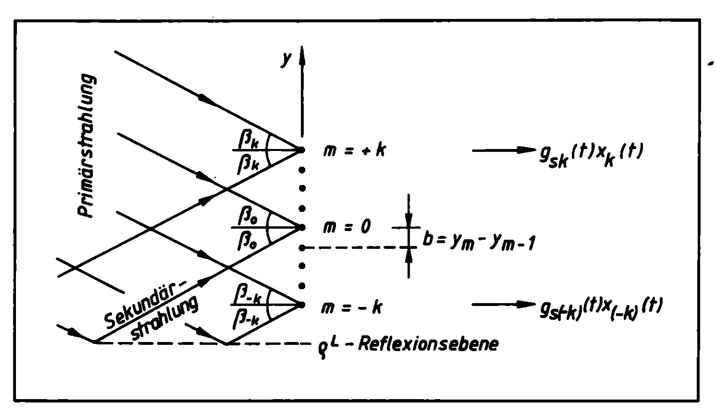


Bild 4. Modellierung des Überlagerungsfalles primärer und sekundärer Strahlungskomponenten

Als einschränkende Bedingung ist dabei anzusehen, daß Amplituden- und Phaseninformationen reflektierter Strahlungskomponenten für den Optimierungsprozeß nicht zugänglich sind [6] und [7].

Voraussetzungsgemäß wird ein m-elementiges Sensorsystem betrachtet, dessen komplex bewertete Elementesignale

$$x_k^L(t) = |x_k^L|\,\mathrm{e}^{j(\omega t + \psi_k)}$$

summierbar sind.

Für den Fall der Symmetriebedingung für die Primärkomponenten des Strahlungsfeldes sowie die Reflexionsfelder

$$x_{Rk}^L(t) = |x_{Rk}^L|\,\mathrm{e}^{j(\omega t + \psi_{Rk})} \tag{16.1}$$

bzw.

$$x_{Rk}^L(t) = \varrho\,|x_k^L|\,\mathrm{e}^{j(\omega t - \psi_k + \Phi_k)} \tag{16.2}$$

mit

$$\psi_k = \frac{2\pi}{\lambda}\,b\sin\beta\,, \qquad \Phi_k = \psi_{pk} - \psi_{Rk}$$

läßt sich das resultierende Elementsignal bilden

$$x_k^L(t) = |x_k^L|\,\mathrm{e}^{j(\omega t + \psi_k)} + \varrho\,|x_k^L|\,\mathrm{e}^{j(\omega t - \psi_k + \Phi_k)}\,. \tag{17}$$

Die Zurückführung des Optimierungsproblems auf die Auswertung eines synthetischen Fehlersignals setzt die Festlegung der Komponentenstruktur dieses Signals voraus. Analog zu (10) läßt sich ein Fehlersignal $\sigma(t)$ als Funktional der Steuerfunktion $s(t) = f[\sigma(t)]$ bilden.

Aus

$$\sigma(t) = r(t) - g\sum_{k=-m}^{m} g_{sk}x_k(t) \tag{18}$$

mit

$$r(t) = \sum_{k=-m}^{m} g_{rk}x_k(t) \tag{19}$$

folgt

$$\sigma(t) = \sum_{k=-m}^{m} g_{rk}x_k(t) - g\sum_{k=-m}^{m} g_{sk}x_k(t) \tag{20}$$

als Summe der Signalkomponenten $r(t)$ sowie $x(t)$ gemäß dem Modellierungsansatz nach Bild 4.

Die Lösung des Extremwertproblems

$$\operatorname{grad}_g\left[\sum_{i=1}^{p} \sigma(t_i)\;\sigma^*(t_i)\right] = 0 \tag{21}$$

ermöglicht analog dem Verfahren (13) die Ermittlung der Variationsfunktionen g_{opt}

$$g_{opt} = g \Big|_{\mathrm{grad}_g \left[\sum_{i=1}^{p} \sigma(t_i)\, \sigma^*(t_i) \right] = 0} \,. \tag{22}$$

Mit den Optimierungsalgorithmen (13) und (22) ist damit das sensorische Modell des Erfassungsmechanismus elektromagnetischer Strahlungsfelder der eingangs formulierten Eigenschaften für den Fall interferierender Konfigurationen synthetisierbar, d. h., die Konzeptionsbedingungen multifunktionaler adaptiver Erkennungssysteme sind analytisch festgelegt. Untersuchungen zu möglichen Mechanismen der Referenzsignalerzeugung sind für den Überlagerungsfall bandgespreizter Primärsignale [8] und [9] sowie schmalbandiger primärer FSK-Signale [8] und [10] mit breitbandigen Interferenzsignalen bekannt.
Zuzuordnen sind die Änderungen der Signalparameter des Primärsignals als Funktional der Übertragungscharakteristik des Referenzsignalkreises insbesondere für den Fall willkürlich belegbarer Sensorfelder. Die betrags- und phasenmäßigen Signalparameteränderungen für den Fall willkürlich belegbarer Sensorfelder mit überlagerter unkorrelierter Rauschsignalkomponente sind über die analytischen Untersuchungen [8] und [11] ableitbar.
Setzt man ein Sensorfeld der Vorschrift gemäß Bild 5 voraus, läßt sich ein komplexer Eingangssignalvektor der Form

$$\begin{bmatrix} x_1^L(t) \\ x_2^L(t) \\ x_3^L(t) \\ \vdots \\ x_{n-1}^L(t) \\ x_n^L(t) \end{bmatrix} = \begin{bmatrix} |x_1^L|\, e^{j[\omega t + \psi_1]} \\ |x_2^L|\, e^{j\left[\omega t + \psi_2 - \frac{2\pi}{\lambda_0} \Delta l \sin\alpha\right]} \\ |x_3^L|\, e^{j\left[\omega t + \psi_3 - \frac{4\pi}{\lambda_0} \Delta l \sin\alpha\right]} \\ \vdots \\ |x_{n-1}^L|\, e^{j\left[\omega t + \psi_{n-1} - 2(n-2)\frac{\pi}{\lambda_0} \Delta l \sin\alpha\right]} \\ |x_n^L|\, e^{j\left[\omega t + \psi_n - 2(n-1)\frac{\pi}{\lambda_0} \Delta l \sin\alpha\right]} \end{bmatrix} \tag{23}$$

bilden.
Die Änderungen der Signalparameter des Primärsignals als Funktional der Übertragungscharakteristik des Referenzsignalkreises mit dem Referenzsignal $r(t)$ sind in Abhängigkeit der Felddimension des Sensorfeldes darstellbar.
Mit

$$r(t) = \frac{X^T W}{|X^T W|}\, a_r\, e^{-j\psi_r} \tag{24}$$

a_r Referenzkreisdämpfung; ψ_r Phasenkonstante des Referenzkreises
folgt

$$\begin{aligned} k = 1&: \lim_{t\to\infty} s(t) = |s_1^L|\, f_{\Delta 1}\, e^{j[\omega_0 t + \psi_s]} \\ k = 2&: \lim_{t\to\infty} s(t) = |s_2^L|\, f_{\Delta 2}\, e^{j[(\omega_0 + \Delta\omega_1)t + \psi_s]} \\ k = 3&: \lim_{t\to\infty} s(t) = |s_3^L|\, f_{\Delta 3}\, e^{j[(\omega_0 + \Delta\omega_2)t + \psi_s]} \\ &\vdots \\ k = n-1&: \lim_{t\to\infty} s(t) = |s_{n-1}^L|\, f_{\Delta n-2}\, e^{j[(\omega_0 + \Delta\omega_{n-2})t + \psi_s]} \\ k = n&: \lim_{t\to\infty} s(t) = |s_n^L|\, f_{\Delta n-1}\, e^{j[(\omega_0 + \Delta\omega_{n-1})t + \psi_s]} \end{aligned} \tag{25}$$

mit $f_{\Delta 1}, \dots, f_{\Delta n-2}, f_{\Delta n-1} = \cos\psi_r$,

$$\Delta\omega_1 = \frac{2 + \dfrac{2\sigma_N^2}{x^2}}{1 - \dfrac{2\sigma_N^2}{x^2}}, \dots, \Delta\omega_{n-1} = \frac{2 + \dfrac{n\sigma_N^2}{x^2}}{1 + \dfrac{2\sigma_N^2}{x^2}} \,. \tag{26}$$

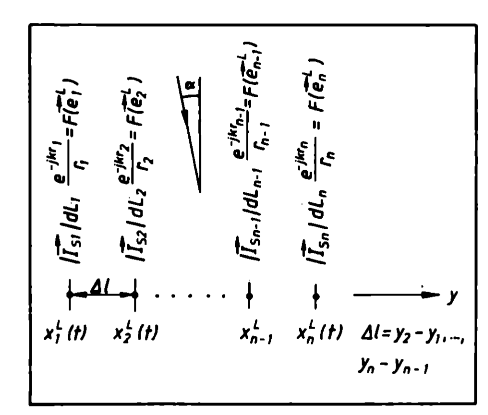


Bild 5. Strukturierung des Sensorfeldes

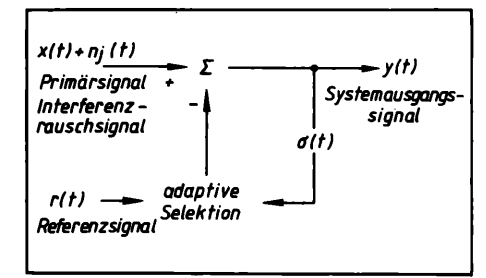


Bild 6. Systemkonzeption adaptiver Rauschsignalunterdrückung

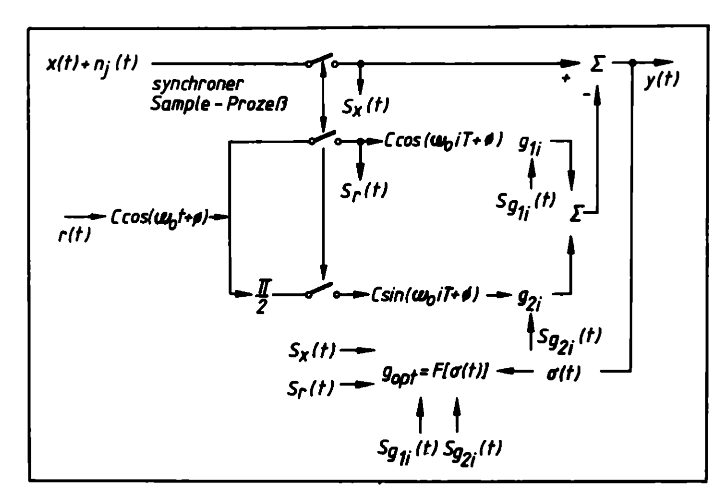


Bild 7. Systemlösung zweiwertiger Adaptionszyklen

3. Adaptive Rauschsignalunterdrückung

Zugehörige Untersuchungen zur Optimierung des Signal-Rausch-Verhaltens unter Berücksichtigung sowohl der internen als auch externen Quellen überlagerter Rauschkomponenten wurden in [18] und [19] durchgeführt.
Lösungsalgorithmen der adaptiven Rauschsignalunterdrückung für den Fall der Überlagerung des Primärsignals mit interferierenden Rauschsignalkomponenten sind auf der Grundlage der Untersuchungen in [12] bis [14] formulierbar. Die konzeptionelle Möglichkeit zur Lösung des Problems der adaptiven Rauschsignalunterdrückung wird im Bild 6 dargestellt.
Der Adaptionsmechanismus ist dabei auf den Lösungsalgorithmus des Extremwertproblems [15] und [16] zurückführbar. Die Systemlösung für den Fall eines zweiwertigen Adaptionsmechanismus wird in [17] abgeleitet (Bild 7).
Konzeptionelle Lösungsmöglichkeiten der Schaltungssynthese phasendrehender Netzwerke sind über die Untersuchungen [20] bis [29] ableitbar. Insbesondere die Verfahren zur Phasensteuerung mit magnetostatischen Oberflächenwellen [28] und [29] sind von aktueller Bedeutung für planare Lösungen von Sensorproblemen.

Schlußbemerkungen

Da im Vergleich zum sichtbaren Spektralbereich das Mikrowellenspektrum die Möglichkeit der Durchstrahlung dielektrischer Ob-

jekte beinhaltet und da für den Fall elektrisch vollständig leitender Objekte die Anwendbarkeit der Beugungstheorie zur Beschreibung der Objektinformationen gegeben ist, sind die diskutierten Adaptionsmechanismen von Mikrowellensensorsystemen als Erweiterungsmöglichkeiten technischer Erkennungssysteme zur Erzielung von Anpassungsmechanismen der Sensor- und Systemfunktionale aufzufassen.

Der allgemeine Fall eines elektromagnetischen Strahlungsfeldes ist durch die Überlagerung von der Kohärenz abweichender Felder zu Interferenzfeldern, in denen konventionelle Methoden der Strahlungsdetektion versagen können, gekennzeichnet. Mit der Aufgabenformulierung der Strahlungsdetektion für den Fall der Überlagerung des Strahlungsfeldes von der Kohärenz abweichender Felder als korrelierte sowie unkorrelierte Komponenten zum primären Detektionssignal ist der Erkennungsvorgang als Erfassung gestörter oder ungestörter Wertemuster mit der Eigenschaft stochastischer Veränderlichkeit festgelegt. Für allgemeine Sensorkonzeptionen des Hochfrequenzspektrums elektromagnetischer Strahlungsfelder wird a priori die Existenz ungestörter ebener Wellenfelder vorausgesetzt, d. h., die Überlagerung der Einzelfelder der Strahlungs- bzw. Feldsensoren eines Detektionssystems wird als Interferenz gleichphasiger Partialfelder angenommen.

Postuliert man, daß reale Feldbedingungen mit der Ausbildung von Interferenzfeldern infolge der Überlagerung des direkten Strahlungsfeldes mit reflexions- sowie brechungsbedingten indirekten Feldern oder kohärenten Strahlungsanteilen bzw. ausschließlich der Überlagerung indirekter Felder verbunden sind, läßt sich der Zustand des ungestörten Feldes nur als Sonderfall betrachten.

Sensorsysteme mit phasen- und amplitudengleicher Anregung weisen damit eine abhängige Fehlanpassung an den Feldraum des realen Strahlungsfeldes auf.

Für die Ermittlung möglicher Anpassungskonzeptionen von Sensor- und Feldcharakteristik adaptiver Mikrowellensensorsysteme bilden die in diesem Beitrag diskutierten Systembetrachtungen eine Lösungsgrundlage.

Literatur

[1] Barton, P.: Air borne Signal Processing for the Microwave Doppler Landing System. Ph. D. Thesis, University of London, London 1979.

[2] Jaynes, E. T.: Information Theory and Statistical Mechanics. Physical Review 106 (1957) 4, pp. 620—630.

[3] Jaynes, E. T.: Information Theory and Statistical Mechanics. Physical Review 108 (1957) 10, pp. 171—190.

[4] Burg, J. P.: Maximum Entropy Spectral Analysis. 37th Annual International Meeting of the Society of Exploration Geophysicists. Oklahoma City/USA 29. 10. bis 2. 11. 1967.

[5] Burg, J. P.: Maximum Entropy Spectral Analysis. Ph. D. Dissertation, Stanford University of California/USA 1975.

[6] Kesler, J.: A new adaptive antenna for elevation angle estimation in the presence of multipath. Proc. IEEE AP-S Int. Symp., Quebec City. Quebec/Canada, June 1980, pp. 130—133.

[7] Kesler, J.: Adaptive canceller for elevation angle estimation in the presence of multipath. Proc. IEEE 130 (1983) 6, pp. 303—308.

[8] Dicarlo, D. M.: Reference loop phase shift in adaptive arrays. IEEE Trans. Aerosp. Electron Syst. AES-14 (1978), p. 599.

[9] Compton, R. T.: An adaptive array in a spread spectrum communication system. Proc. IEEE 66 (1978), p. 289.

[10] Swarner, W. G.; Berni, A. J.: An adaptive array and angle of arrival estimation device for sensor communications. Electro Science Lab., Dept. of Electrical Engineering, The Ohio State University, Ohio/USA. Rep. 3435-2 (1972).

[11] Dicarlo, D. M.: Reference Loop Phase Shift in an N-Element Adaptive Array. IEEE Trans. Aerosp. Electron. Syst. AES-15 (1979), p. 576.

[12] Widrow, B., u. a.: Adaptive noise cancelling — Principles and applications. Proc. IEEE 63 (1975) 12, p. 1692.

[13] Glover, J.: Adaptive noise cancelling of sinusoidal interference. Ph. D. Dissertation, Stanford Electronics. Lab., Stanford University/USA, 1975.

[14] Glover, J.: Adaptive noise cancelling applied to sinusoidal interferences. IEEE Trans. Acoust., Speech, Signal Processing, ASSP-25 (1977) 6, p. 484.

[15] Widrow, B.; Hoff, Mc. E.: Adaptive switching circuits. IRE WESCON Conv. Rec. 4 (1960) 96.

[16] Widrow, B., u. a.: Stationary and nonstationary learning characteristics of the LMS adaptive filter. Proc. IEEE 64 (1976) 8, p. 1151.

[17] Widrow, B.; Kenneth, M. D.; Richard, P. G.; Newman, W. C.: Signal Cancellation Phenomena in Adaptive Antennas. IEEE Trans. on Antennas and Propagation AP-30 (1983) 3, p. 469.

[18] Cheng, D. K.; Tseng, F. I.: Signal-to-noise ratio maximization for receiving arrays. IEEE Trans. Antennas and Propagation (1966) Nov., pp. 792—794.

[19] Cummins, J.; Delisle, G. Y.; Pelletier, M.: Internal Noise in the SNR Optimization of Receiving Arrays. Proc. IEEE 65 (1977) Aug., pp. 1196—1197.

[20] White, J. F.: High Power, p-i-n Diode Controlled, Microwave Transmission Phase Shifters. IEEE Trans. on Microwave Theory and Techniques, MTT-13 (1965) March, pp. 232—242.

[21] Opp, F. L.; Hoffman, W. F.: Design. of Digital Loaded-Line Phase-Shift Networks for Microwave Thin-Film Applications. IEEE Trans. on Microwave Theory and Techniques. MTT-16 (1968) July, pp. 162 to 168.

[22] Möller, E.: Analyse und Entwurf von digitalen Mikrowellenphasenschiebern für phasengesteuerte Antennen. Dissertation, RWTH Aachen, Aachen/BRD 1970.

[23] Rusch, J.: Fremdgesteuerte elektronische Charakteristikschwenkung von Antennen und Raumflugkörpern. Forschungsbericht W 71-46. AEG-Telefunken, Fachbereich Anlagen „Hochfrequenz", Ulm/BRD 1971.

[24] Möller, E.: Digitale Mikrostrip-Ringleitungsphasenschieber aus technischen Bauelementen für X-Band-Frequenzen. NTZ (1972) 5, S. 232—237.

[25] Ulriksson, B.: Continuous Varacter-Diode Phase Shifter with Optimized Frequency Response. IEEE Trans. on Microwave Theory and Techniques MTT-27 (1979) July, S. 650—654.

[26] Bahl, I. J.; Gupta, K. C.: Power Handling Capability of Loaded Line PIN Diode Phase Shifter Circuits. AEÜ 34 (1980) 11, S. 460—464.

[27] Roome, G. T.; Hair, H. A.; Gerst, C. W.: Thin Ferrite Phase-Shifters for integrated Microwave Devices. Journ. of Appl. Physics 38 (1967) 3, pp. 175—176.

[28] Edenhofer, P.; Krug, J.: Untersuchungen zur Speisung und Phasensteuerung von planaren Strahlergruppen mit magnetostatischen Oberflächenwellen. AEÜ 36 (1982) 10, S. 409—411.

[29] Krug, J.; Edenhofer, P.: Planare Strahleranordnung mit magnetostatischer Oberflächenwellenspeisung. NTG-Fachberichte, Bd. „Antennen" (1982), S. 52—57.

[30] Lin, H.-C.: Spatial Correlations in Adaptive Arrays. IEEE Trans. Antennas and Propagat. AP-30 (1982), p. 212.

[31] Bakhru, K.; Torrieri, D. J.: The Maximum Algorithm for Adaptive Arrays and Frequency-Happing Communications. IEEE Trans. Antennas and Propagat. AP-32 (1984), p. 919.

[32] Gupta, I. J.; Ksienski, A. A.: Dependence of Adaptive Array Performance on Conventional Array Design. IEEE Trans. Antennas and Propagat. AP-30 (1982), p. 549.

[33] Compton, R. T.; Huff, R. J.; Swarner, W. G.; Ksienski, A. A.: Adaptive arrays for communication systems — An overview of research at The Ohio State University. IEEE Trans. Antennas and Propagat. AP-24 (1976), p. 599.

msr 8840

L. Zipser, J. Labude

Planarer akustischer Abluftfeuchtesensor

0. Einleitung

Auf der Basis fluidischer Oszillatoren wurde ein robuster Gasfeuchtesensor entwickelt, der auch unter extremen Meßbedingungen, z. B. in der heißen und verschmutzten Abluft industrieller Trockner zuverlässig und genau arbeitet. Der Sensor ist in Verbindung mit einem Mikrorechner als robustes Feuchtemeßgerät oder zusammen mit einem Regler zur energieoptimalen Regelung industrieller Trocknungsprozesse nutzbar.

1. Meßprinzip

Primäre Meßwandler fluidisch-akustischer Gasfeuchtesensoren sind fluidische Oszillatoren [1]. Deren Frequenzverhalten ist im akustischen Arbeitsbereich mit der Gleichung

$$f = A \cdot c + B(\Delta p) \tag{1}$$

beschreibbar. A ist eine Oszillatorkonstante, die von der Geometrie des Oszillators abhängt. Die Funktion $B(\Delta p)$ berücksichtigt die Abhängigkeit der Frequenz f von der Druckdifferenz zwischen der Auslauf- und Einlaufdüse des Oszillators. Ferner ist c die Schallgeschwindigkeit in dem Gas, mit dem der Oszillator betrieben wird.
Ist das betreibende Gas ein Gemisch aus Wasserdampf (Index W) und Luft (Index L) mit dem Masseverhältnis (Feuchtegrad)

$$X = \frac{m_W}{m_W + m_L}, \tag{2}$$

gilt unter bestimmten Voraussetzungen für die Schallgeschwindigkeit [2]

$$c = \sqrt{\frac{\varkappa(X)}{M(X)} RT} = \sqrt{\left(\frac{X c_{pW} + (1-X) c_{pL}}{X c_{vW} + (1-X) c_{vL}}\right)\left(\frac{X}{M_W} + \frac{(1-X)}{M_L}\right) RT}\,. \tag{3}$$

$\varkappa$ Verhältnis der spezifischen Wärmen c_p/c_v; M Molmasse der Gase; R allgemeine Gaskonstante; T absolute Temperatur des Gasgemisches

Bei bekannten Größen A, $B(\Delta p)$, T, c_p und M läßt sich mit (1) bis (3) aus der Oszillatorfrequenz f der Feuchtegrad X des Gasgemisches bestimmen.

2. Funktion des Abluftfeuchtesensors

Bei Messung des Feuchtegrades X mit üblichen fluidischen Oszillatoren verfälschen veränderliche Oszillatorkonstanten $A_{m,r}$, Druckdifferenzen Δp und Temperaturen T die Meßergebnisse.
In [3] wurde ein planarer fluidisch-akustischer Gasfeuchtesensor vorgeschlagen, bei dem die Wirkungen der veränderlichen Größen $A_{m,r}$, Δp und T kompensiert sind (Bild 1). Er kann vorzugsweise zur Messung der Abluftfeuchte in industriellen Trocknungsanlagen eingesetzt werden. Bei diesem Sensor werden fluidisch-akustische Oszillatoren verwendet, die in einem definierten Druckbereich dominant akustisch arbeiten, so daß ihre Frequenz f von der Druckdifferenz unabhängig ist, d. h., es gilt $B(\Delta p) \to 0$.
Zur Korrektur der störenden Einflüsse veränderlicher Ablufttemperaturen T und instabiler Oszillatorkonstanten A hat der Abluftfeuchtesensor neben dem Meßoszillator (Index m) einen geometrisch gleichen Referenzoszillator (Index r).

Meßzustand:

Meß- und Referenzoszillator sind an einen gemeinsamen Ejektor angeschlossen. Dieser saugt im Meßzustand durch den Meßoszillator die zu analysierende feuchte Abluft und durch den Referenzoszillator trockene Luft aus dem Druckluftnetz. Die Temperaturen von Ab- und Referenzluft werden durch thermische Kopplung beider Oszillatoren angeglichen, so daß $T_m = T_r$ gilt. Gleichzeitig ist der Referenzoszillator als Temperatursensor nutzbar.
Der Quotient der Frequenzen beider Oszillatoren ist dann eine Funktion der Oszillatorkonstanten $A_{m,r}$ und des Feuchtegrades X:

$$\delta f(X, A) = \frac{f_m}{f_r} = \frac{A_m}{A_r}\sqrt{\frac{\varkappa(X)}{M(X)}\frac{M_L}{\varkappa_L}} = \frac{A_m}{A_r}\sqrt{\left(\frac{X c_{pW} + (1-X) c_{pL}}{X c_{vW} + (1-X) c_{vL}}\right)\left(\frac{X}{M_W} + \frac{(1-X)}{M_L}\right)\left(\frac{M_L}{\varkappa_L}\right)}\,. \tag{4}$$

Die Oszillatorkonstanten A_m und A_r sollen gleich groß sein und sich in (4) aufheben. Jedoch rufen Fertigungstoleranzen und Verschmutzungen geometrische Unsymmetrien zwischen Meß- und Referenzoszillator hervor, die zu ungleichen Konstanten A_m und A_r und damit zu Meßfehler verursachenden Verschiebungen der Sensorkennlinie führen. Diese Verschiebungen lassen sich mit einer im Abluftfeuchtesensor eingebauten Nullpunktkontrolleinrichtung [4] unter Prozeßbedingungen messen und anschließend korrigieren.

Nullpunktkontrolle:

Zur Kontrolle des Nullpunktes des Abluftfeuchtesensors werden Meß- und Referenzoszillator mit gleichem Gas betrieben, z. B. wird im Referenzoszillator trockene Luft gegen Abluft ausgetauscht:

$$\frac{\varkappa_L}{M_L} \to \frac{\varkappa(X)}{M(X)}. \tag{5}$$

Meß- und Referenzoszillator schwingen jetzt unter identischen Bedingungen. Folglich werden im Idealfall gleich großer Oszillatorkonstanten $A_m = A_r$ wegen (4) und (5) bei beliebigen Feuchtegraden X Sollquotienten

$$\delta f_S(X) = \frac{A_m}{A_r} = 1 \tag{6}$$

gemessen. Dieser Sollquotient repräsentiert den idealen Nullpunkt der Sensorkennlinie.
In der Praxis weichen die Oszillatorkonstanten mehr oder weniger voneinander ab, d. h. $A_m \neq A_r$. Folglich treten bei der Nullpunktkontrolle Istquotienten

$$\delta f_I(X, A) = \frac{A_m}{A_r} = 1 + \frac{\Delta A}{A} = \delta f_S(X) + \Delta(\delta f(A)) \tag{7}$$

auf. Der Summand $\Delta(\delta f(A)) \ll 1$ entspricht mit sehr guter Näherung der Nullpunktabweichung der X/f-Kennlinie des Abluftfeuchtesensors. Er wird im betrachteten Kontrollzustand des Sensors angezeigt.

Dr. sc. techn. LOTHAR ZIPSER (44) lernte Elektromechaniker und studierte von 1962 bis 1968 Elektrotechnik an der TU Dresden. 1976 Promotion A an der TU Dresden. 1986 Promotion B zu Problemen der Modellierung fluidischer Oszillatoren und ihrer Anwendung bei der Sensorentwicklung. Zur Zeit wissenschaftlicher Mitarbeiter am Zentralinstitut für Kybernetik und Informationsprozesse (ZKI) der AdW der DDR, Institutsteil Dresden.

Dr.-Ing. JOACHIM LABUDE (47) studierte von 1960 bis 1966 Wärmetechnik an der TH Karl-Marx-Stadt. Von 1966 bis 1973 wissenschaftlicher Assistent am Wissenschaftsbereich Wärmetechnik der Sektion Verarbeitungstechnik der TH Karl-Marx-Stadt. 1972 Promotion A. Seit 1973 im VEB Textilmaschinenbau Gera, stellvertretender Direktor Forschung und Entwicklung.

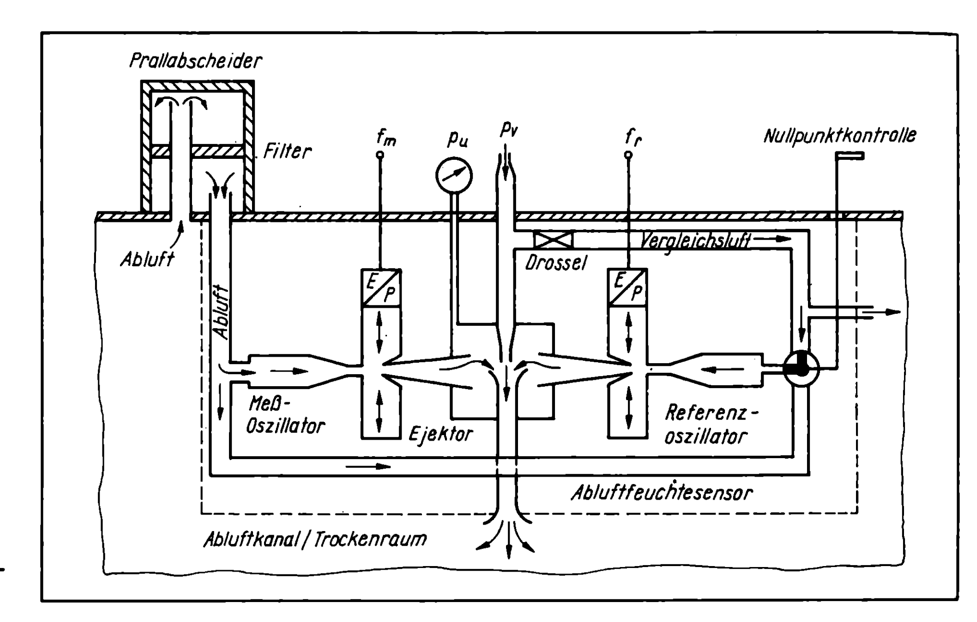


Bild 1. Funktionsschema des planaren Abluftfeuchtesensors

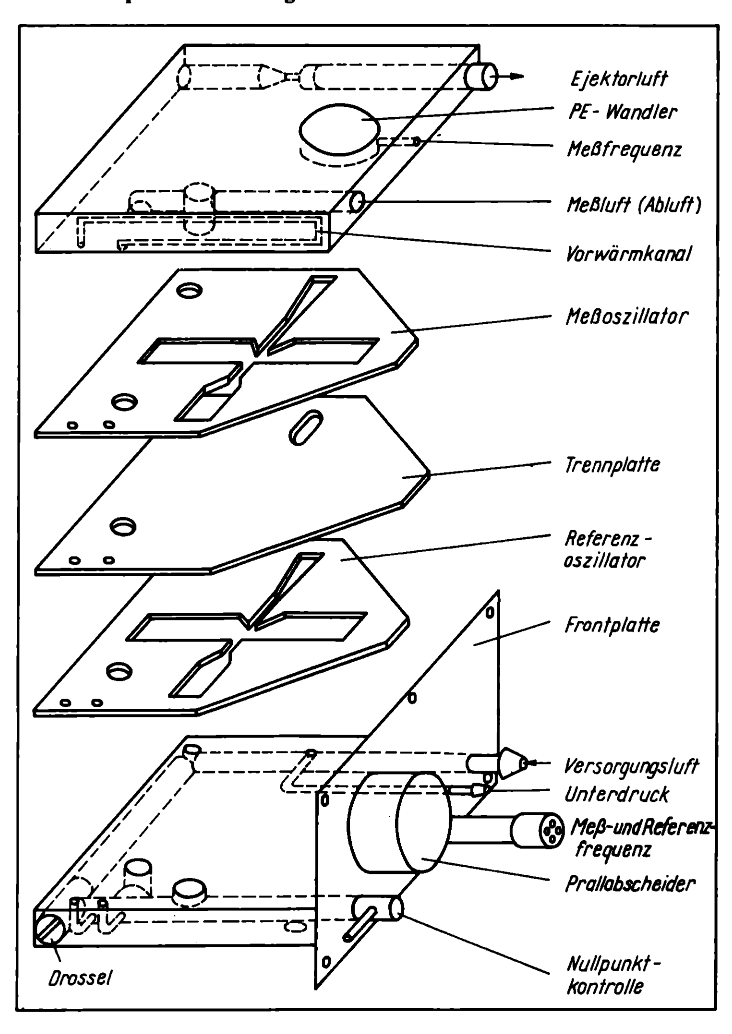
In welchen Zeitabständen erneute Nullpunktkontrollen durchzuführen sind, hängt vom Verschmutzungsgrad der Abluft ab und ist individuell festzulegen. Werden erhebliche Nullpunktabweichungen $-0{,}01 > \Delta(\delta f(A)) > 0{,}01$ festgestellt, ist eine Reinigung des Abluftfeuchtesensors erforderlich.

Nullpunktkorrektur:
Im Meßzustand bei beliebigen Feuchtegraden X gemessene Istquotienten $\delta f_\mathrm{I}(X, A)$ sind bei Nullpunktverschiebungen meßfehlerbehaftet. Diese Istquotienten sind durch Subtraktion des vorher ermittelten Wertes $\Delta(\delta f(A))$ auf die korrigierten Sollquotienten

$$\delta f_\mathrm{S}(X) = \delta f_\mathrm{I}(X, A) - \Delta(\delta f(A)) \qquad (8)$$

und über (4) auf den zu bestimmenden Feuchtegrad X zurückführbar. Durch diese Korrektur werden alle Meßfehleranteile, die durch geometrische Unsymmetrien von Meß- und Referenzoszillator oder durch deren Verschmutzung entstehen, beseitigt.

Bild 2. Explosionsdarstellung des Abluftfeuchtesensors



3. Aufbau des Abluftfeuchtesensors

Der fluidisch-akustische Abluftfeuchtesensor hat einen planaren Aufbau. Er besteht aus:

- Grundplatte mit Druckwandler
- Referenzoszillatorplatte
- Trennplatte
- Meßoszillatorplatte
- Deckplatte mit Druckwandler
- Frontplatte mit Prallabscheider und Bedienelementen.

Die Platten sind durch Schrauben verbunden (Bild 2). Damit kann der kompakte Sensor einfach demontiert und gereinigt werden. Zum Messen wird der Sensor in einen Abluftkanal oder in eine Trockenkammer eingebaut. Die Abluft wird vor dem Eintritt in den Sensor in einem Prallabscheider [5] mit Filter gereinigt. Der Prallabscheider kann ohne Ausbau des Sensors von außen gesäubert werden.
Die Frequenzen f_m und f_r der Oszillatoren werden mit piezoelektrischen Wandlern aufgenommen, mit einem Mikrorechner oder Mikrorechnerregler ausgewertet und Feuchte- bzw. Stellsignale bereitgestellt.

4. Meßsignalerfassung mit Mikrorechner

Der Mikrorechner realisiert folgende Funktionen:

Frequenzmessung:

Meß- und Referenzfrequenz werden gleichzeitig nach dem Periodendauerverfahren gemessen. Das ermöglicht schnelle und genaue Frequenzmessungen auch bei niedrigen Signalfrequenzen. Die Anzahl der gemessenen Perioden ist von 1024 bis 8192 in 4 Schritten einstellbar.

Meßwertberechnung:

Aus den Einzelfrequenzen f_m und f_r werden das Frequenzverhältnis δf berechnet und eventuelle Nullpunktverschiebungen korrigiert. Die Beträge der Nullpunktverschiebungen werden mit der Nullpunktkontrolleinrichtung bestimmt und von Hand in den Mikrorechner eingegeben. Bei der Berechnung des Feuchtegrades X und der Temperatur T der Abluft bleibt der Nichtlinearitätsfehler im Meßbereich $X = 0 \ldots 1000$ g/kg und $T = 60 \ldots 200$ °C unter 0,5%. Neben dem Feuchtegrad X lassen sich auch andere Feuchtemaße berechnen und anzeigen.

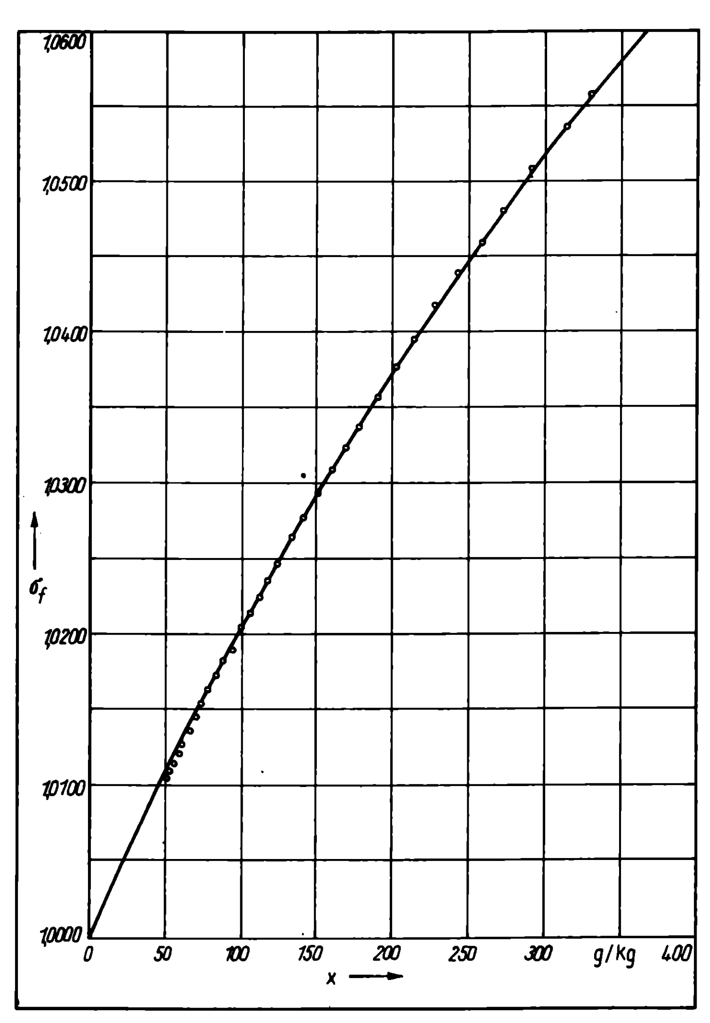


Bild 3. Experimentelle und theoretische Kennlinie des Abluftfeuchtesensors

Anzeige:

Der Mikrorechner hat eine Digitalanzeige, mit der wahlweise
- Meßfrequenz f_m
- Referenzfrequenz f_r
- Frequenzverhältnis δf
- Feuchtemeßwerte: $X(X_f)$; X_t; Volumenprozente; Taupunkt τ
- Ablufttemperatur T

angezeigt werden können.

Analogausgabe:

Unabhängig von der gewählten Feuchteeinheit gibt der Mikrorechner einen analogen Meßwert aus, wobei drei Meßbereiche wählbar sind. Eine Balkenanzeige visualisiert gleichzeitig den Analogwert.

Störüberwachung:

Das Programm enthält zahlreiche Überwachungsfunktionen, die im Störfall einen Fehlerkode anzeigen. Damit ist eine schnelle Diagnose von Störungen im Sensor möglich.

5. Kalibrierung des Abluftfeuchtesensors

Die theoretische Kennlinie des Sensors (Bild 3) wurde nach (4) unter der Voraussetzung $A_m = A_r$ berechnet. Die experimentelle Kalibrierung des Sensors erfolgte mit einem Wasserdampf-Luft-Gemisch mit variablem Feuchtegrad X aus einem Generator nach [6]. Die gemessenen Werte bestätigen die theoretische Kennlinie.

6. Anwendung des Abluftfeuchtesensors

Der fluidisch-akustische Abluftfeuchtesensor wurde im Labor und in Trocknern der Textil- und Zellstoffindustrie umfassend erprobt. Dabei hat er seine Eignung als robustes, wartungsarmes und

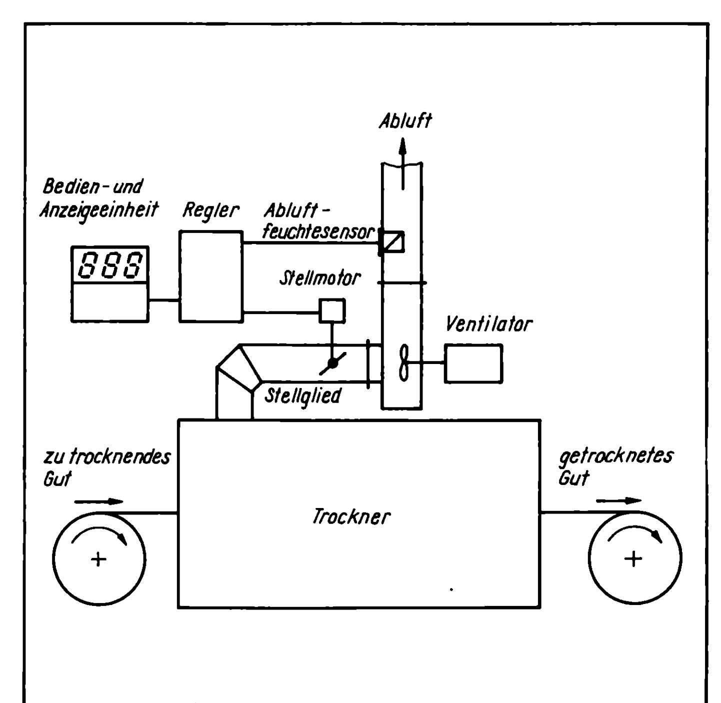


Bild 4. Aufbau der Abluftfeuchtemeß- und -regelanlage

störungsunempfindliches Abluftfeuchtemeßgerät nachgewiesen. Eine Nutzung des Sensors zur energieoptimalen Abluftregelung in industriellen Trocknern kann zu außerordentlich hohen Energieeinsparungen führen:

Beispielsweise wird in Spann-Trocken-Fixiermaschinen der Textilindustrie Stoff von einer hohen Anfangsfeuchte auf die hygroskopische Gleichgewichtsfeuchte getrocknet. Solche Textiltrockner verbrauchen sehr viel Energie und sollten deshalb energieoptimal trocknen. Der Feuchte- und Energiegehalt der Trocknerabluft beeinflußt die Energiebilanz der Trockner entscheidend. Das zur Trocknung erforderliche Abluftvolumen V_{erf} ist so einzustellen, daß der Wärmebedarf und damit die Trocknungskosten (Kosten je kg aufzutrocknendes Wasser) gering bleiben, die Trocknerleistung aber noch ausreichend groß ist. Das wird in den meisten Fällen erreicht, wenn der Feuchtegrad der Abluft bei $X = 100 \dots 150$ g/kg liegt. In der Praxis wird häufig bei viel zu niedrigen Feuchtegraden X, d. h. mit zu großem tatsächlichem Abluftvolumen V_{tats} getrocknet, weil ungenügende Trocknung die Qualität des Gutes mindert, aber notwendige Abluftmeß- und -regeleinrichtungen an den Trocknern (Bild 4) fehlen. Das ist energiewirtschaftlich ungünstig. Nur wenn

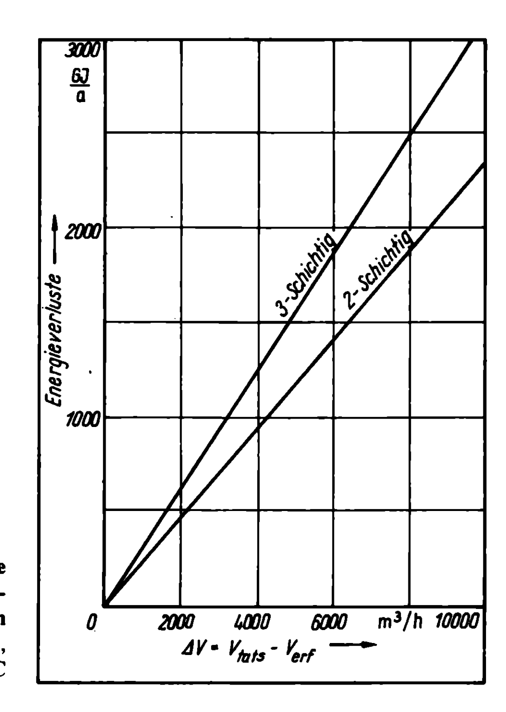


Bild 5. Energieverluste bei extensivem Abluftvolumen
$T_{Abluft} = 130$ °C,
$T_{Raum} = 30$ °C

Tafel. Daten des Abluftfeuchtesensors

Meßbereich	$X = (0 \dots 100, 0 \dots 300$ bzw. $0 \dots 1000)$ g/kg
Meßfehler	$\Delta X < 5$ g/kg
Einschwingzeit	$t < 2$ s (mit Meßwerterfassung)
Einsatztemperatur	$T = 80 \dots 200$ °C (günstig)
	$20 \dots 250$ °C (möglich)
Maße (Sensor)	$95 \times 150 \times 210$ mm³
Masse (Sensor)	3,5 kg
Druckluft	0,04 MPa

für die Abluftvolumen $V_{tats} = V_{erf}$ gilt, arbeitet der Trockner energieoptimal. Im Bild 5 wird gezeigt, welche Energieverluste an industriellen Textiltrocknern auftreten, wenn $V_{tats} > V_{erf}$ ist.
Bei folgenden Betriebszuständen sind durch Einsatz eines Abluftfeuchtesensors mit Regeleinrichtung erhebliche Energieeinsparungen zu erzielen:

— Betrieb des Trockners unterhalb der Nennkapazität
— Aufheizen des Trockners
— Leerlauf während des Betriebes (Rüst- und Wartezeiten, Pausen).

7. Bewertung des Abluftfeuchtesensors

Ein Vergleich zeigte, daß der vorgestellte Abluftfeuchtesensor den auf dem internationalen Markt verfügbaren Abluftfeuchtesensoren in seinen technischen Daten (Tafel) gleichwertig ist und darüber hinaus noch folgende Vorteile aufweist:

— einfacher Aufbau
— Nullpunktkontrolle unter Prozeßbedingungen
— hohe Servicefreundlichkeit
— störungssichere Ausgangssignale
— intelligente Meßsignalerfassung.

Literatur

[1] MEYER, N.; ROHRBACH, CH.: Handbuch für fluidische Meßtechnik. Düsseldorf: VDI-Verlag GmbH 1977.
[2] ZIPSER, L.: Modellierung fluidischer Oszillatoren und ihre Anwendung bei der Entwicklung eines Abluftfeuchtesensors. Dissertation B, TU Dresden 1986.
[3] ZIPSER, L.; DORFMÜLLER, L.; LABUDE, J.: Planarer akustischer Gaszusammensetzungssensor. DDR-PS 237 000 A1 vom 7. 5. 1985, BRD-OS 3 613 125 A1 vom 18. 4. 1986.
[4] ZIPSER, L.; DORFMÜLLER, L.; LABUDE, J.: Nullpunktkontrolleinrichtung für einen akustischen Gaszusammensetzungssensor. DDR-PS 230 083 A1 vom 10. 12. 1984, BRD-OS 3 452 524 A1 vom 2. 12. 1985.
[5] ZIPSER, L.; DORFMÜLLER, L.; LABUDE, J.: Verfahren zur Säuberung verschmutzter heißer Gasgemische. DDR-PS 235 561 A1 vom 21. 3. 1985, BRD-OS 3 608 978 A1 vom 18. 3. 1986.
[6] PLICHTA, J.: Prallstrahlströmung in der Gasfeuchtemeßtechnik. Dissertation A, TH Karl-Marx-Stadt 1984.

msr 8955

E.-G. Woschni

Zur Bedeutung der A-priori-Information speziell in der Meßtechnik

0. Einleitung

Messungen haben das Ziel, unbekannte Signal- bzw. Systemparameter zu ermitteln und damit aus einem Meßobjekt stammende Informationen zu gewinnen. Bereits die Betrachtung der beiden Extremfälle, nämlich, daß entweder gar keine oder die volle Information über das Meßobjekt vor der Messung vorhanden ist, zeigt, daß offensichtlich die A-priori-Information, d. h. die vor der Messung bereits vorliegende Information über das Meßobjekt, von ausschlaggebender Bedeutung ist. Im ersten Fall ist eine Messung nicht möglich, da keinerlei Aussagen, z. B. zur Art des Meßgerätes oder zum erforderlichen Meßbereich, vorliegen, im zweiten Fall ist eine Messung überflüssig.
Der in der Praxis gegebene Tatbestand liegt zwischen diesen beiden Extremfällen. Dabei sind oft die erreichbare Meßgenauigkeit und damit der durch die Messung erreichbare Informationsgewinn umso größer, je mehr A-priori-Information vorliegt, da dann eine bessere Anpassung des Meßgerätes an die Meßaufgabe möglich ist.
Im Beitrag soll gezeigt werden, wie die Lösung des Meßproblems von der vorhandenen A-priori-Information abhängt und welche Konsequenzen sich ergeben, z. B. die Bedeutung von Abschätzungen und Näherungsverfahren gerade in der Meßtechnik.
Möglichkeiten zur Anwendung der Gedanken der robusten Regelungen in der Meßtechnik werden angedeutet.

Prof. Dr.-Ing. habil. Dr.-Ing. E. h. EUGEN-GEORG WOSCHNI (60) studierte von 1947 bis 1951 an der TH Dresden. Danach wissenschaftlicher Assistent und Oberassistent an der TH Dresden. 1953 Promotion A, 1956 Habilitation. Seit 1957 Professor an der TH bzw. TU Karl-Marx-Stadt. 1984 Ehrenpromotion an der TU Dresden. Vorstandsmitglied der WGMA.

1. Grundsätzliche Zusammenhänge

In der Meßtechnik besteht meist ein linearer Zusammenhang zwischen Ausgangs- und Eingangsgrößen y und x, so daß die lineare Systemtheorie angewendet werden darf [1] und [2]. Für die ideale und reale Ausgangsgröße ermittelt man mit den Störgrößen z

$$y_{id} = Op_{id}\{x\}\,, \tag{1a}$$

$$y_{real} = Op_{real}\{x, z\}\,. \tag{1b}$$

Die linearen Operationen können der Fourier-Transformation unterzogen werden und führen zu den Frequenzgängen

$$G_{id}(j\omega) = F\{Op_{id}\}\,, \tag{2a}$$

$$G_{real}(j\omega) = F\{Op_{real}\}\,. \tag{2b}$$

Der mittlere quadratische Fehler $\overline{\varepsilon^2(t)}$ beträgt damit nach Bild 1 [1] [2] und [3]

$$\overline{\varepsilon^2(t)} = \overline{\varrho^2(t)} + \overline{z^2(t)} = \\ = \int_{-\infty}^{+\infty} S_{xx}(\omega)\,|G_{id}(j\omega) - G_{real}(j\omega)|^2\,d\omega + \int_{-\infty}^{+\infty} S_{zz}(\omega)\,d\omega\,. \tag{2c}$$

Dabei wird eine additive, unkorrelierte Störung angenommen, so daß die beiden Fehleranteile, der dynamische Fehler $\overline{\varrho^2(t)}$ und der störungsbedingte Fehler $\overline{z^2(t)}$, addiert werden dürfen. Anderenfalls ist die Korrelation zu berücksichtigen [3].
In der Regelungstechnik ist die Eingangsgröße x als Führungsgröße w gegeben und die Ausgangsgröße gesucht. In der Meßtechnik dagegen liegt der umgekehrte Fall vor: Die Kenntnis der Eingangsgröße x ist das Ziel der Messung, sie ist nicht gegeben. Trotzdem ist in beiden Fällen das Problem zu lösen, $G(j\omega)$ so zu bestimmen, daß

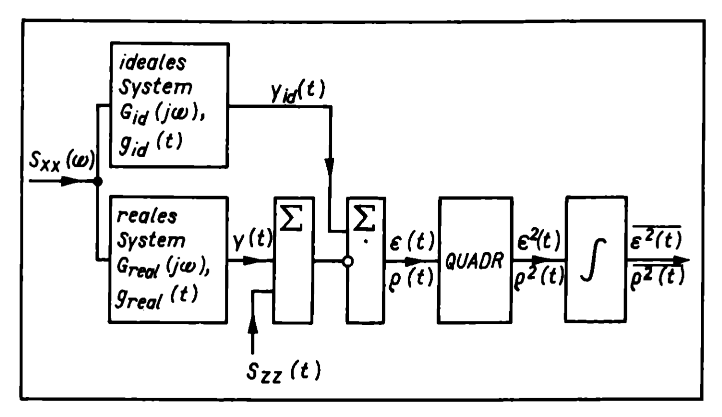


Bild 1. Zur Berechnung des mittleren quadratischen Fehlers

ein zulässiger Fehler nicht überschritten wird. Man erkennt, daß dieses Problem in der Regelungstechnik bei bekannter Eingangsgröße grundsätzlich lösbar ist, in der Meßtechnik dagegen nicht.
Der einzige Ausweg besteht daher darin, zunächst Annahmen über die Eingangsgröße aufgrund von abschätzbaren Informationen über den zu messenden Prozeß für die zu erwartende Meßgröße abzuleiten und dann nach der Messung festzustellen, ob diese Annahmen soweit richtig waren, daß der Fehler klein genug bleibt. Man erkennt die große Bedeutung der A-priori-Information und wegen der ohnehin notwendigen Annahmen über die Eingangsgröße die Wichtigkeit sowie die Berechtigung, die gerade in der Meßtechnik Abschätzverfahren und Näherungsbetrachtungen zukommt. Dies soll anschließend anhand einiger auch für die Praxis bedeutsamer Beispiele erläutert werden.

2. Beispiele

2.1. Auswahl von Meßgeräten

Besonders problematisch ist die Messung von kurzen Impulsen, wie sie beispielsweise als Belastungsspitzen im Maschinenbau (u. a. bei Verbrennungsmotoren) und in der Elektrotechnik auftreten. Ein Fehler bei der Messung der Impulshöhe führt zur Zerstörung des betreffenden Gerätes. Dabei liegt der kuriose Fall vor, daß sich kleine Fehler oft schlimmer auswirken als große Meßfehler: Aufgrund der Sicherheitszuschläge bei der Dimensionierung hält das Bauelement bei Meßfehlern von einigen % bis 10% zunächst und übersteht oft noch die Nullserie. Wegen der dauernden relativ geringfügigen Überlastung bricht dann u. U. nach Jahren, wenn die Geräte bereits in Serie gefertigt und vielleicht sogar schon exportiert sind, an allen Maschinen das gleiche Teil entsprechend der nach der Wöhler-Kurve begrenzten Zahl der Lastwechsel bis zum Bruch. Ein grober Meßfehler dagegen wird bereits beim Versuchsmuster erkannt.
Für eine richtige Messung der Spitzenbelastung benötigt man eine A-priori-Information der Impulsbreite Δt. Wie Bild 2 zeigt, wird die Übergangsfunktion für die Abschätzung durch eine Rampenfunktion mit der Einschwingzeit t_E ersetzt. Diese Einschwingzeit hängt nach dem Abtasttheorem mit der Grenzfrequenz entsprechend der Beziehung $f_g = 1/(2t_E)$ zusammen. Damit ergeben sich entsprechend Bild 2 je nach dem Verhältnis von $\Delta t/t_E$ die gestrichelten bzw. strichpunktierten Ausgangsfunktionen. Man erkennt, daß bei zu geringer Impulsbreite erhebliche Meßfehler auftreten können.
Für die Praxis ergeben sich daraus folgende Hinweise:

- Man kann der Messung erst dann vertrauen, wenn auch ein dynamisch besseres Meßgerät denselben Wert liefert.
- Eine Ausgangsfunktion mit gegenüber der Vorderflanke wesentlich verlängerter Rückflanke deutet auf einen Meßfehler hin. In diesem Fall kann die Größe des Meßfehlers aus dem Verhältnis der Anstiegs- und Abklingzeit abgeschätzt und gegebenenfalls korrigiert werden. Beträgt beispielsweise die Abklingzeit das 4fache der Anstiegszeit, so wird nur etwa 1/4 der tatsächlichen Belastung wiedergegeben.

Damit erhält man folgende einfache Beziehung zur Abschätzung der richtigen Impulshöhe H im Vergleich zur gemessenen H_{gem}

$$H \approx H_{gem} \cdot t_E/\Delta t\,. \qquad (3a)$$

Eine genauere Betrachtung hat die Tatsache zu berücksichtigen, daß sich die Übergangsfunktion aus e-Funktionen zusammensetzt [1]. Eine Annäherung durch ein System mit Verzögerung 1. Ordnung und Ausgleich führt zur Darstellung im Bild 3 mit den eingezeichneten Möglichkeiten zur Ermittlung der Einschwingzeit $t_E = 3T$ aus der Zeitkonstante T. Ist der Impuls nicht ideal, so entsteht eine Antwortfunktion mit einer Krümmung in der Umgebung des Nullpunktes. In diesem Fall ist die Zeitkonstante an der steilsten Stelle der Kurven zu ermitteln [1]. Schließlich kann eine Rückrechnung auf den tatsächlichen Verlauf der Eingangsgröße $x_{real}(t)$ bei einem System 1. Ordnung rechnerisch oder grafisch nach der Beziehung

$$x_{real}(t) = y(t) + T\,\frac{dy(t)}{dt} \qquad (3b)$$

erfolgen [1].

2.2. Abschätzung von Fehlern infolge Anti-Aliasing-Filterung

Ein weiteres Beispiel stellt die Abschätzung der Fehler dar, die durch die Anti-Aliasing-Filterung hervorgerufen werden (Cut-off-Fehler). Nach Bild 4 werden durch die Tiefpaßfilterung zur Erfüllung des Abtasttheorems die Frequenzen oberhalb der halben Abtastfrequenz $\omega_A/2$ abgeschnitten. Dadurch entsteht nach [1] und [4] ein mittlerer quadratischer Fehler

$$\overline{\varepsilon^2} = 2\int_{\omega_A/2}^{\infty} S_{xx}(\omega)\,d\omega\,. \qquad (4)$$

Zur Berechnung des Fehlers ist folglich die A-priori-Kenntnis der spektralen Leistungsdichte des Signals $S_{xx}(\omega)$ erforderlich. Da diese vor der Messung i. allg. nicht bekannt ist, nimmt man für eine

Bild 2. Abschätzung der Auswirkung des Meßgerätes auf die Messung am Beispiel der Impulsmessung

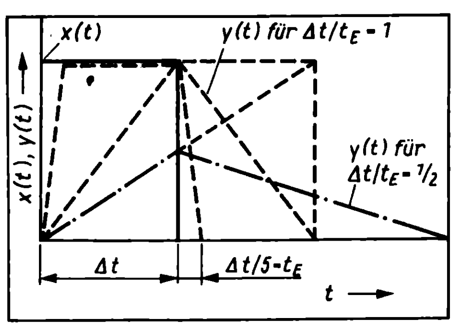


Bild 3. Gegenüber Bild 2 verbesserte Abschätzung

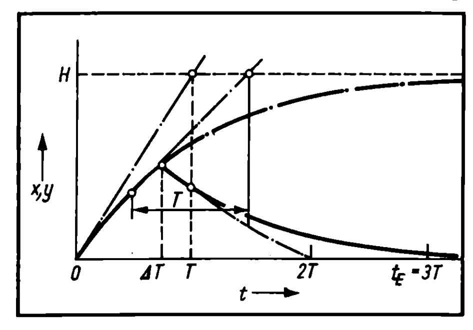
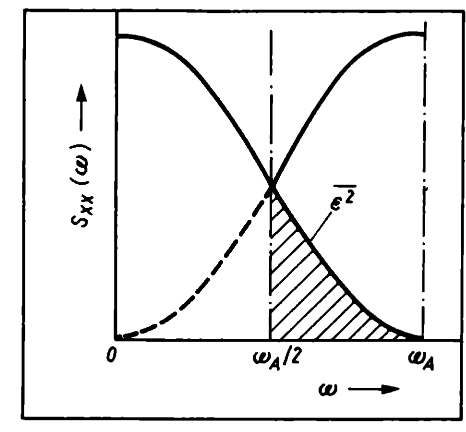


Bild 4. Abschätzung der Aliasing-Fehler



Abschätzung meist ein oberhalb einer Frequenz ω_0 abfallendes Spektrum an:

$$S_{xx}(\omega) = \frac{S_{xx0}}{1 + (\omega/\omega_0)^2} \,. \tag{5a}$$

Der mittlere quadratische Fehler lautet dann

$$\overline{\varepsilon^2} = 2 S_{xx0} \omega_0 \left[\frac{\pi}{2} - \arctan \frac{\omega_A}{2\omega_0} \right]. \tag{5b}$$

Unter der Voraussetzung eines nach (5a) abfallenden Spektrums muß also nunmehr zumindest die für das Spektrum charakteristische Frequenz ω_0 bekannt sein. Die Abschätzung kann zur Bestimmung des Zusammenhanges zwischen einer Verminderung des Fehlers und einer Erhöhung der Abtastfrequenz benutzt werden: Eine Reihenentwicklung liefert für $\omega_A/2\omega_0 > 1$ mit $\arctan x \approx \approx \frac{\pi}{2} - \frac{1}{x}$

$$\overline{\varepsilon^2} = 4 S_{xx0} \omega_0^2 / \omega_A \,, \tag{5c}$$

d. h., zur Verringerung des Cut-off-Fehlers um eine Größenordnung ist eine Erhöhung der Abtastfrequenz um ebenfalls eine Größenordnung erforderlich. Dies ist z. B. beim Einsatz von Mikrorechnern mit größerer Wortbreite zu beachten [5].

Diese Abschätzung kann auch für die Aliasing-Fehler durchgeführt werden [4], wobei nach Bild 4 der mittlere quadratische Fehler einschließlich des Aliasing-Fehlers wegen der gleichen Fläche zum Doppelten des Wertes nach (4) bzw. (5b) abgeschätzt werden kann, d. h.

$$\overline{\varepsilon^2_{ges}} = 4 S_{xx0} \omega_0 \left[\frac{\pi}{2} - \arctan \frac{\omega_A}{2\omega_0} \right]. \tag{5d}$$

Es sei an dieser Stelle vermerkt, daß diese Gleichung nur eine Abschätzung darstellt, da die für eine Addition der beiden Fehleranteile notwendige Voraussetzung fehlender Korrelation genaugenommen nicht erfüllt ist.

2.3. *Informationsbetrag und Kanalkapazität*

Im folgenden soll ebenfalls eine Näherungsbetrachtung vorgenommen werden, die zu einer anschaulichen Deutung der Ergebnisse der exakten Theorie führt, wie sie bei SHANNON zu finden sind [6]:

Bei einem mittleren quadratischen Fehler $\overline{\varepsilon^2(t)}$ und einer maximal möglichen Ausgangsleistung P_y kann man maximal $\hat{m}_p = P_y/\overline{\varepsilon^2(t)} + 1$ Leistungsstufen unterscheiden. Daraus ergeben sich $\hat{m}_a$ Amplitudenstufen

$$\hat{m}_a = \hat{m}_p^{1/2} = [P_y/\overline{\varepsilon^2(t)} + 1]^{1/2} \,. \tag{6a}$$

Folglich benötigt man zur Speicherung eines Wertes $H_0 = {}^2\log \hat{m}_a$ binäre Speicherplätze. H_0 stellt den Informationsbetrag in bit für den günstigsten Fall, z. B. Gaußscher Wahrscheinlichkeitsverteilung bei Leistungsbegrenzung, dar [1].

Da nach dem Abtasttheorem maximal $1/t_E = 2f_g$ Werte je Sekunde auftreten können, erhält man für den maximalen Informationsfluß I_{max} in bit/s

$$I_{max} = 2f_g \cdot {}^2\log \hat{m}_a = f_g \cdot {}^2\log \hat{m}_p = f_g \cdot {}^2\log \left[\frac{P_y + \overline{\varepsilon^2(t)}}{\overline{\varepsilon^2(t)}} \right]. \tag{6b}$$

Der dynamische Fehleranteil läßt sich als systematischer Fehler korrigieren, so daß sich mit dem dann noch verbleibenden störungsbedingten Fehler die Beziehung für die Kanalkapazität C_t ergibt:

$$C_t = f_g \, {}^2\log \frac{P_y + P_z}{P_z} \,. \tag{6c}$$

Dies ist die von SHANNON angegebene Beziehung [6].

Durch Einführung des logarithmischen Dämpfungsmaßes in dB erhält man hieraus die zugeschnittene Größengleichung

$$C_t = \frac{1}{3} B \cdot \frac{P_y}{P_z} \,. \tag{6d}$$

C_t in bit/s; B in Hz; P_y/P_z in dB

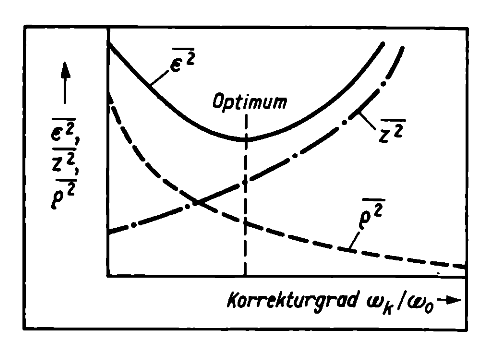


Bild 5. Verlauf der Teilfehler und des Gesamtfehlers in Abhängigkeit von der Bandbreite des Korrektursystems (überwiegender dynamischer Fehler)

Für die in der Praxis auftretenden Signal-Rausch-Verhältnisse von 20 ... 90 dB ergibt sich die Faustformel

$$C_t \approx (6 \dots 30)\, B \,. \tag{6e}$$

2.4. *Optimalfilterung*

Mit einem nach der im Bild 1 additiv hinzutretenden Störung in Reihe geschalteten Korrekturfilter kann man sowohl den dynamischen Fehleranteil $\overline{\varrho^2(t)}$ als auch den störungsbedingten $\overline{z^2(t)}$ beeinflussen, wobei die Verminderung des einen Anteils durch eine Erhöhung des jeweils anderen erkauft werden muß. Dabei stellt sich bei der sog. optimalen Filterung ein Minimum des mittleren quadratischen Gesamtfehlers $\overline{\varepsilon^2(t)}$ ein (Bild 5). Nach der Optimalfiltertheorie [2] und [3] erhält man für den optimalen Frequenzgang des Gesamtsystems

$$G_{ges\,opt}(j\omega) = G_{id}(j\omega) \frac{S_{xx}(\omega)}{S_{xx}(\omega) + S_{zz}(\omega)} \tag{7a}$$

bzw. für den Frequenzgang des in Reihe geschalteten Korrekturfilters

$$G_{korr}(j\omega) = \frac{G_{id}(j\omega)}{G_{real}(j\omega)} \frac{S_{xx}(\omega)}{S_{xx}(\omega) + S_{zz}(\omega)} \,. \tag{7b}$$

Diese Filter sind nicht realisierbar, so daß das tatsächliche System durch entsprechende Variationen von (7a, b) ermittelt werden muß [2] und [3].

Man entnimmt auch hier aus den Beziehungen, daß die Kenntnis der Leistungsspektren sowohl des Nutz- als auch des Störsignals Voraussetzung für die Bemessung des Filters ist. Da diese Kenntnis nicht vorliegt, müssen auch hier Abschätzungen unter Benutzung der vorhandenen A-priori-Information verwendet werden.

2.5. *Berücksichtigung der „Robustheit"*

Zur Realisierung einer Optimierung bei nicht vollständiger A-priori-Information gewinnen neuerdings Gedanken, die in der Theorie für robuste Regelungen entwickelt wurden, an Bedeutung [7]. Diesem Ansatz liegt die Idee zugrunde, die zugelassenen Fehler durch obere Schranken zu beschreiben. Dieses Kriterium entspricht damit vom Prinzip her demjenigen, das bereits bei der klassischen Tschebyschev-Approximation für Filtercharakteristiken angewendet wird. Bei den „robusten Reglern" wird verlangt, daß trotz der Unsicherheiten infolge nicht genügender Kenntnisse des Prozeßmodells, einschließlich vorhandener Nichtlinearitäten oder zeitlicher Änderungen der Parameter, auch für Mehrgrößen-Regelungssysteme mit Sicherheit — im Gegensatz zur unscharfen Betrachtungsweise, bei der nur die Einhaltung einer vorgegebenen Wahrscheinlichkeit verlangt wird — die Einhaltung der vorgegebenen Fehlerschranken gewährleistet ist. Damit lassen sich auch Systeme einbeziehen, deren Verhalten sich in gewissen Grenzen zeitlich ändert bzw. nichtlinear ist. Die Anwendung dieser Gedanken auch in der Meßtechnik ist wegen der im Prinzip gleichen Forderungen bezüglich der Einhaltung von Fehlerschranken interessant und naheliegend, wurde jedoch bisher noch nicht im einzelnen vorgenommen.

Welche Probleme dabei in der Meßtechnik auftreten, sei am Beispiel der Auswirkungen von Parameteränderungen bei korrigierten Meßsystemen erläutert: Bekanntlich verlangt die Korrektur von Systemen 1. Ordnung, z. B. Temperaturaufnehmern, durch ein nachgeschalte-

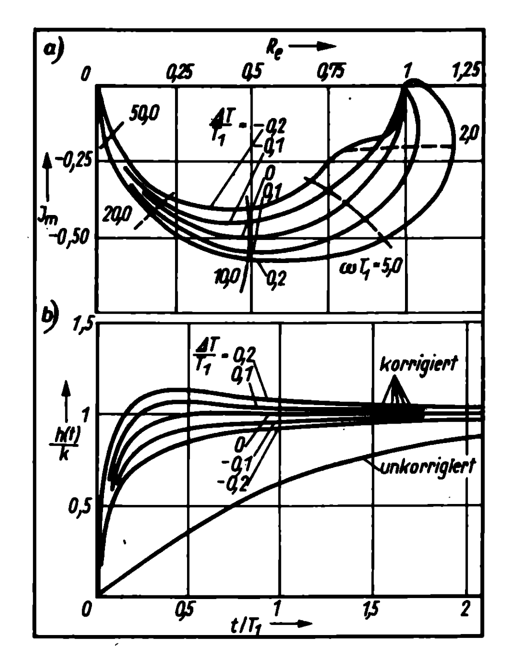


Bild 6. Kennfunktionen eines um den Faktor 10 korrigierten Systems 1. Ordnung bei Parameteränderungen (nach [8])
a) Ortskurve;
b) Übergangsfunktion

tes Vorhaltglied mit Verzögerung 1. Ordnung die Einhaltung der Bedingung, daß die Nullstelle des Korrektursystems mit dem Pol des Originalsystems zusammenfällt, d. h., daß die Zeitkonstanten T_1 und T_k gleich sind. Diese Bedingung wird verletzt, wenn — praktisch unvermeidlich — Parameteränderungen der Zeitkonstante T_1 um ΔT_1 auftreten. Die Bilder 6a und b zeigen die Auswirkungen auf Ortskurve und Übergangsfunktion eines mit dem Faktor 10 korrigierten Systems [8], d. h., die Zeitkonstante des korrigierten Systems beträgt $T_1/10$. Man erkennt, daß bereits relativ geringe Parameteränderungen zu erheblichen Veränderungen der Kennfunktionen führen. Da nach [8] die Auswirkungen von Parameteränderungen und damit die durch sie hervorgerufenen Fehler grundsätzlich um so größer werden, je höher der Korrekturgrad sowie die Grade der zu korrigierenden Systeme sind, können die Forderungen nach Robustheit erhebliche Einschränkungen bezüglich der Korrekturmöglichkeiten mit sich bringen.

Ausblick

Der im Abschnitt 2.3. angegebene Zugang von der Informationstheorie her liefert mit dem maximalen Informationsfluß bzw. der Kanalkapazität ein Gütekriterium, in das zwei technische Kennwerte, nämlich der Fehler und die Grenzfrequenz als Kennwert für das dynamische Verhalten, eingehen.
Eine Erweiterung auf ein allgemeines Gütekriterium $Q.K.$ könnte nach dem gleichen Prinzip vorgenommen werden, wobei neben dem Informationsfluß I nach (6b) weitere Einflußgrößen c_r durch Gewichtsfunktionen λ_r berücksichtigt werden

$$Q.K. = \lambda_0(I) + \sum_r \lambda_r(c_r) . \tag{8a}$$

Diese allgemeine Formulierung enthält z. B. als Sonderfall auch die Produktdarstellung, die zum Informationsfluß nach (6b) führt

$$Q.K. = {}^2\log\left[(\hat{m}_a)^{f_g} \prod_r c_r^{\alpha_r}\right] = f_g\,{}^2\log \hat{m}_a + \sum_r \alpha_r\,{}^2\log c_r\,, \tag{8b}$$

nämlich mit $\lambda_0 = 1$; $\lambda_r(c_r) = \alpha_r\,{}^2\log c_r$.

Im einfachsten Fall werden in der Praxis oft Gewichtsfaktoren γ_r und Exponenten α_r vorgesehen

$$Q.K. = I + \sum_r \gamma_r c_r^{\alpha_r}\,, \tag{8c}$$

da eine Einbeziehung semantischer und pragmatischer Aspekte in ein allgemeines Kriterium aufgrund einer entsprechend erweiterten Informationstheorie nicht möglich ist.
Ein typisches Beispiel für eine weitere nichttechnische Einflußgröße c_r sind die Kosten. Sie enthalten semantische und pragmatische Aspekte, so daß auch die hier übliche Einführung von „Straffunktionen" keine grundsätzliche und eindeutige Lösung des Problems der Einbeziehung derartiger Einflußgrößen ergibt [1].

Literatur

[1] Woschni, E.-G.: Informationstechnik. Signal, System, Information. 3. Aufl. Berlin: VEB Verlag Technik 1988 und Heidelberg: Dr. A. Hüthig Verlag GmbH 1988.
[2] Krauss, M.; Woschni, E.-G.: Meßinformationssysteme. 2. Aufl. Berlin: VEB Verlag Technik 1975, Heidelberg: Dr. A. Hüthig Verlag GmbH 1975 und Moskau: MIR 1975.
[3] Schlitt, H.: Systemanalyse für regellose Vorgänge. Berlin/Göttingen/Heidelberg: Springer-Verlag 1960.
[4] Woschni, E.-G.: Minimising Aliasing Errors of Sensors with Digital Output. J. Phys. Sci. Instrum. E. 20 (1987), S. 119—124.
[5] Woschni, E.-G.: Probleme im Zusammenhang mit einer Erhöhung der Verarbeitungsbreite von Mikroprozessoren bei ON-LINE-Verarbeitung. Preprints 32. Internationales Wissenschaftliches Kolloquium, TH Ilmenau 1987, H. 1, S. 237—240.
[6] Shannon, C. E.: A Mathematical Theory of Communication. BSTJ 27 (1948), S. 379.
[7] Lunze, J.: Ansätze zum Entwurf linearer Regelungssysteme auf Robustheit. msr, Berlin 29 (1986) 6, S. 245—251.
[8] Woschni, E.-G.: Parameterempfindlichkeit in der Meßtechnik, dargestellt an einigen typischen Beispielen. msr, Berlin 10 (1967) 4, S. 124 bis 130.

msr 8890

Der Instrumentationsverstärker — ein hochwertiger Meßverstärker

0. Einleitung

Seit 1985 stellt der *VEB Keramische Werke Hermsdorf* (KWH) drei Varianten analoger Hybridschaltkreise, sog. Instrumentationsverstärker, her. Der Beitrag begründet die Notwendigkeit des Einsatzes dieser Verstärker zur Realisierung qualifizierter Meßaufgaben im µV- und mV-Bereich unter Störeinfluß.

1. Vom Operationsverstärker zum Instrumentationsverstärker

In der Meßtechnik und bei der Datenerfassung kommt es häufig vor, daß lange Leitungen zu den Signaleingängen des Verstärkers führen und daher Stör- und Rauschsignale auftreten. In der Praxis entstehen zwischen zwei geerdeten Punkten immer Spannungen, deren Differenz mehrere Volt betragen kann. Es handelt sich dabei um Komponenten der Netzfrequenz und ihrer Harmonischen sowie um Induktionsspitzen.
Bei einfachen Verstärkern liegt ein Signaleingang auf Masse. Soll das Signal einer mittelbar oder unmittelbar an Netzerde liegenden Quelle erfaßt werden, ist die direkte Verbindung eines Signaleingangs mit Netzerde zu vermeiden. Sonst dringt die Störspannung stark in den Meßstromkreis ein. Bild 1 zeigt einen einfachen netzversorgten Verstärker. Bei diesem bestehen durch die Streukapazität des Netztrafos zwischen Masse und Netzerde die Impedanzen Z1 und Z2, wobei letztere praktisch die Einkopplung der Störspannung $U_{\text{stör}\,2}$ ermöglicht. Diese Impedanzen, die aus einer Streukapazität parallel zum Isolationswiderstand bestehen, differenzieren nämlich stark. Um die Störung einzuschränken, kann der bekannte Differenzverstärker mit einem Operationsverstärker eingesetzt werden (Bild 2). Für den Betrag der Spannungsverstärkung gilt

$$|V_U| = \frac{R2}{R1} \quad \text{mit} \quad R1 = R1' \quad \text{und} \quad R2 = R2'.$$

Auch hier ergeben sich zu beachtende Impedanzen $Z1 \gg R1$ und $Z2 \gg R2$ gegen Netzerde (Bild 3). $R1$ und $R2$ sind die Zuleitungswiderstände. Es gilt für die Eingangsspannung [1]

$$U_i = U_{\text{mess}} + \left(\frac{R1}{Z1} - \frac{R2}{Z2}\right) U_{\text{stör}}.$$

Die Ausgangsspannung ist

$$U_0 = V_U U_i + V_{Gl} U_{\text{stör}} = V_U U_{\text{mess}} + V_U \left(\frac{R1}{Z1} - \frac{R2}{Z2}\right) U_{\text{stör}} + V_{Gl} U_{\text{stör}},$$

wobei V_{Gl} die Gleichtaktverstärkung des Differenzverstärkers ist. Die Gleichung drückt aus, daß die Gleichtaktstörspannung $U_{\text{stör}}$ teilweise in eine Serientaktstörspannung umgewandelt wird, die voll verstärkt am Ausgang auftritt. Wird der Klammerausdruck zu Null, kann keine Serientaktstörspannung entstehen. Dieser Zustand kann praktisch nicht erreicht werden. In $R1$ ist auch der Innenwiderstand der Quelle eingeschlossen. Da wegen $R1$, $R2 \ll Z1$, $Z2$ der Wert in der Klammer aber stets sehr gering ist, kann der Fehler in akzeptablen Grenzen gehalten werden.
Nicht vernachlässigen darf man den Fehler durch den Gleichtaktanteil. Die Gleichtaktverstärkung des herkömmlichen Differenzverstärkers wird durch die Paarung der wirksamen Gegenkopplungswiderstände stark beeinflußt. Unvermeidbare Störgrößen sind die Streukapazitäten C_{streu} und der Signalquellen-Innenwiderstand R_i (Bild 4), der oft nicht zeitkonstant oder mit einer reaktiven Komponente behaftet ist. In der Praxis sind daher Abweichungen nicht zu vermeiden. Ein weiterer schwerwiegender Nachteil des Differenzverstärkers besteht darin, daß entweder ein hoher Eingangswiderstand (bestimmt durch $R1$, $R1'$) oder eine hohe Verstärkung erreicht werden können. Würde man beides anstreben, wären die durch den (temperaturabhängigen) Offsetstrom und die Streukapazitäten im Zusammenhang mit den hochohmigen Widerständen verursachten Fehler unvertretbar.
Mit dem Instrumentationsverstärker (Bild 5) können die genannten Schwierigkeiten vermieden sowie weitere Vorteile erzielt werden.

2. Aufbau des Instrumentationsverstärkers

Ein Instrumentationsverstärker besteht im Prinzip aus einem gewöhnlichen Differenzverstärker, dem zwei nichtinvertierende Operationsverstärker vorgeschaltet sind, die den hohen Eingangswiderstand realisieren. Der Spannungsabfall an R entspricht der Differenzeingangsspannung. An $R1$ und $R1'$ treten dazu proportionale, entgegengesetzt gepolte Spannungsabfälle auf. Daher wird zwischen den Ausgängen eine verstärkte Spannung erhalten, wobei gleichzeitig noch die Gleichtaktspannung der Eingänge auftritt. Der Differenzverstärker beseitigt diese und addiert die gegensinnigen Signale.
Durch die Differenzschaltung der Eingangsverstärker ergibt sich der Vorteil, daß hinsichtlich der Driftgrößen und des Rauschens nur die Differenzwerte zum Tragen kommen. Im Hinblick auf wirksame Gleichtaktunterdrückung und gutes Signal-Rausch-Verhältnis ist es günstig, den überwiegenden Anteil der Gesamtverstärkung mit

Bild 1. Störung eines einfachen Meßverstärkers (z. B. Operationsverstärker in invertierender oder nichtinvertierender Schaltung)

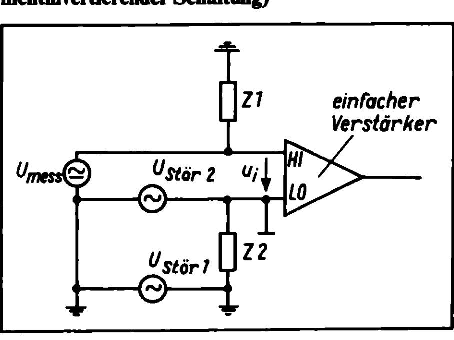


Bild 2. Differenzverstärker, mit einem Operationsverstärker aufgebaut

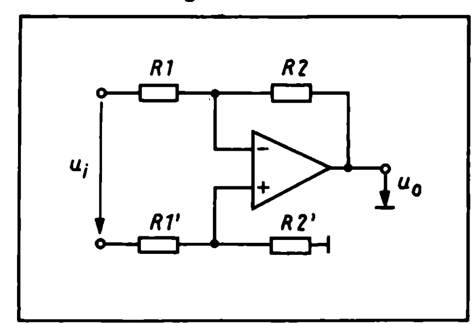
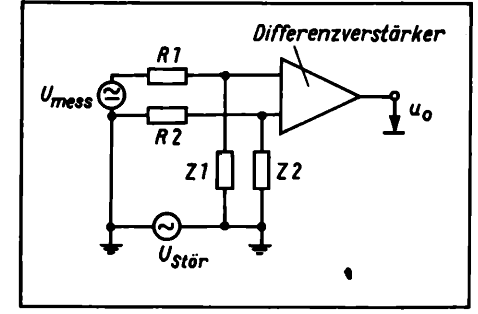


Bild 3. Störung eines Differenzverstärkers



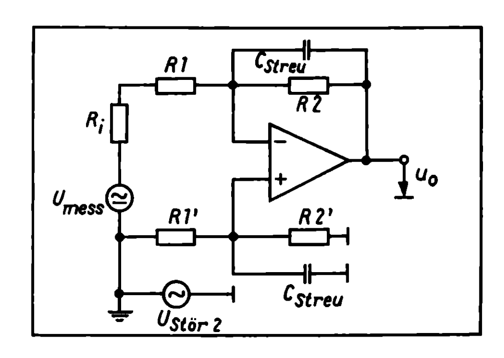


Bild 4. Einfacher Differenzverstärker mit relevanten Einflußgrößen für Gleichtaktansteuerung bzw. -verstärkung

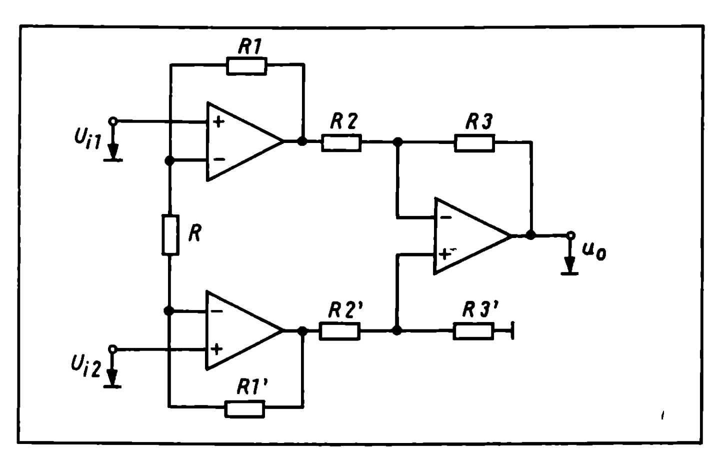


Bild 5. Grundaufbau des Instrumentationsverstärkers

den Eingangsverstärkern zu realisieren. Für die Gesamtverstärkung gilt:

$$V_U = \left(1 + \frac{2R1}{R}\right)\frac{R3}{R2},$$

$$V_U = \frac{U_0}{U_{i1} - U_{i2}}.$$

Die Festlegung erfolgt mit R und damit ohne Beeinflussung anderer Eigenschaften des Instrumentationsverstärkers.
Instrumentationsverstärker können mit Einzeloperationsverstärkern, in Hybridtechnik oder als monolithisch integrierter Schaltkreis aufgebaut werden. R wird entweder extern zugeschaltet, oder man integriert einige hochkonstante Widerstände (z. B. in Dünnfilmtechnik).

3. Instrumentationsverstärker aus dem KWH

Nachfolgend werden die wichtigsten Informationen zu den Typen aus KWH-Produktion gegeben. Sie haben ein 25poliges Metall-Glas-Hermetikgehäuse, das den Einsatz auch in prozeßnaher Umgebung gestattet. Eine ausführliche Beschreibung erfolgte in [2].

3.1. Instrumentationsverstärker INA 30

Der INA 30 ist besonders für die Verstärkung kleiner von Meßwertgebern kommender Gleichspannungen in mikrorechnergesteuerten Meßwerterfassungsanlagen geeignet. Durch die Ausnutzung der technischen Möglichkeiten der Hybridtechnik und die Anwendung des Funktionsabgleichs werden sehr gute Werte bei den Parametern Genauigkeit und Linearität, Offsetspannung und Temperaturkoeffizient erreicht. Die Frequenzkompensation erfolgt intern, die Verstärkung wird mit einem externen Widerstand eingestellt.

3.2. Schneller Instrumentationsverstärker INA 52

Der INA 52 ist ähnlich dem INA 30 aufgebaut. Auch hier werden bei den bereits genannten Parametern sehr gute Werte erzielt. Durch die Integration von hochpräzisen, sehr stabilen Widerständen ist die Einstellung der Verstärkungen 10, 100 und 1000 durch äußere Brücken möglich. Weiterhin kann eine Einstellung durch die Beschaltung mit einem externen Widerstand im Bereich von 5 bis 1000 erfolgen.

3.3. Anpassungs-Instrumentationsverstärker TTA 20

Der TTA 20 findet als Anpassungsschaltkreis für Widerstandsgeber Verwendung. Er wandelt die dort anstehenden Meßspannungen in für die Verarbeitung in Meßwerterfassungssystemen geeignete Spannungen um.
Der Schaltkreis enthält eine spannungsgesteuerte Konstantstromquelle, einen Instrumentationsverstärker und einen Anpassungsverstärker. Durch die Konstantstromeinspeisung erhält man ein widerstandsproportionales Spannungssignal, das vom Instrumentationsverstärker fehlerfrei verstärkt und vom Operationsverstärker auf den benötigten Pegel eingestellt wird. Dabei kann die Verstärkung unter Nutzung interner Widerstände eingestellt werden.

4. Applikationshinweise

Der Einsatz des Instrumentationsverstärkers beschränkt sich im Gegensatz zum Operationsverstärker auf reine Verstärkeranwendungen. Die höheren Genauigkeitsanforderungen führen aber zu erweiterten Beschaltungsmöglichkeiten, die der Hersteller in seinen Datenblättern angibt. Darüber hinaus ergeben sich besonders bez. der Eingänge und des Differenzverhaltens Eigenschaften, die Beachtung verdienen.
Ausführliche Hinweise dazu können [3] entnommen werden.
In der Praxis muß man überlegen, ob ein Eingangsspannungsschutz erforderlich ist. Wird ein Instrumentationsverstärker dazu eingesetzt, Störungen durch Spannungen zwischen den Signalleitungen und Erde zu beseitigen, können bei langen Leitungen hohe Spannungen eingestreut werden. Daneben gibt es Schaltungen, wo hohe elektrostatische Spannungen vorkommen. In diesen Fällen läßt sich die Schutzschaltung nach Bild 6 einsetzen, die die Eingangsspannungen auf die Höhe der Betriebsspannung begrenzt.
Instrumentationsverstärker lassen sich sehr gut an Quellen anschließen, die „schwimmend" betrieben werden. In diesem Fall (Bild 7a) darf aber nicht versäumt werden, die Eingangsruheströme über einen Widerstand R zuzuführen. Erfolgt die kapazitive Abtrennung einer Gleichkomponente zur Gewinnung des Eingangssignals, ist dieser

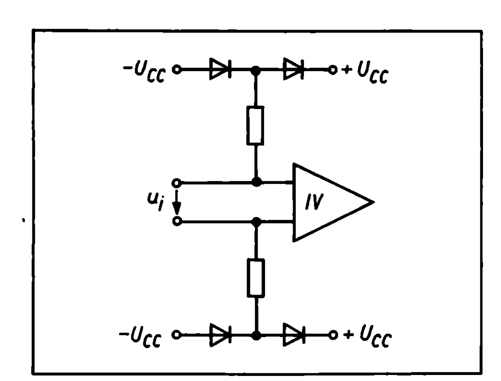
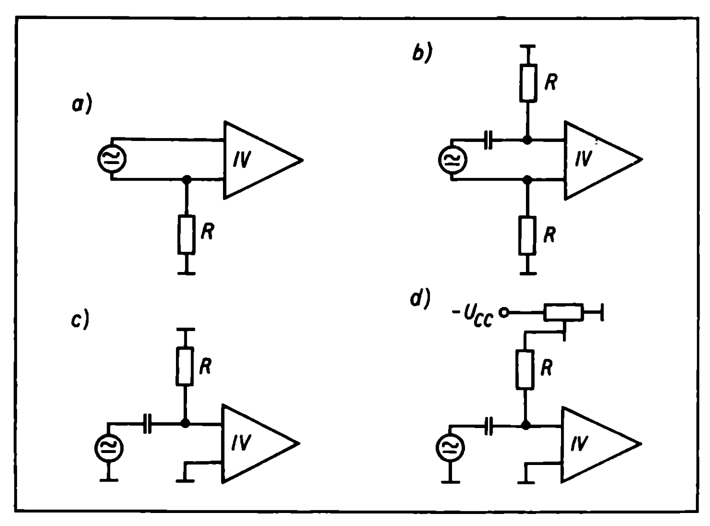


Bild 6. Eingangsschutz eines Instrumentationsverstärkers
Dioden: 4 SAY 30, 32

Bild 7. Varianten der Eingangsruhestromzuführung



bei beiden Eingängen vorzusehen (Bild 7b). Die Ableitwiderstände ändern aber mehrere Eigenschaften der Schaltung. Sie überbrücken den Gleichtakteingangswiderstand bzw. zusätzlich den Differenzeingangswiderstand. Bei Bild 7a fließt der Eingangsruhestrom des oberen Eingangs durch die Quelle und kann bei genügend hohem Innenwiderstand einen ungewollten Spannungsabfall erzeugen. Die Summe der Eingangsruheströme sorgt an *R* für eine Gleichtaktansteuerung. Eine solche liegt auch bei Bild 7b vor, jedoch ist sie nur etwa halb so hoch. Dafür kommt es aber durch die Differenz der Eingangsruheströme zu einer Differenzansteuerung, die durch Abgleich eines Widerstandes vermieden werden kann. Wird der Instrumentationsverstärker bei kapazitiver Ankopplung der Quelle unsymmetrisch betrieben (Bild 7c), ist ebenfalls eine Zuführung des Eingangsruhestroms für den „heißen" Eingang erforderlich. Dabei sorgt der an *R* entstehende Spannungsabfall für eine relativ hohe ungewollte Ansteuerung. Kann diese den Instrumentationsverstärker aus dem linearen Arbeitsbereich bringen oder ist am Ausgang keine kapazitive Abtrennung der Fehlspannung vorgesehen, ist die Schaltung nach Bild 7d einzusetzen.

Literatur

[1] SICHLA, F.: Meßgeräte mit Operationsverstärkern. electronica 230. Berlin: Militärverlag 1986.

[2] SICHLA, F.: Instrumentationsverstärker INA 30, INA 52 und TTA 20. rfe, Berlin 37 (1988) 6, S. 375—377.

[3] SICHLA, F.: Instrumentationsverstärker. rfe, Berlin 37 (1988) 10, S. 633 und 634.

msr 8984

Ing. F. SICHLA, Zentrum für Forschung und Technologie des VEB Elektroprojekt und Anlagenbau, Stammbetrieb des VEB Kombinat Automatisierungsanlagenbau, Rhinstr. 100, Berlin 1140 (bei Manuskripteingang, inzwischen ist der Autor freischaffend)

Modellierung der Kommunikationsprozesse im automatisierten Produktionssystem

0. Einleitung: Kommunikationsprozesse im automatisierten Produktionssystem

Die auf der Mikroelektronik beruhende Computertechnik sowie die Kommunikationstechniken und -technologien bilden eine der Hauptvoraussetzungen zur Schaffung einer durchgängigen Automatisierung des Produktionsprozesses. Die zunehmende innerbetriebliche Computerisierung durchdringt nicht nur die Bereiche der Produktionsdurchführung (CAM: Computer Aided Manufacturing) und vorgeschalteter Prozesse, wie Entwicklung, Konstruktion und Projektierung (CAD: Computer Aided Design) und technologische Vorbereitung der Produktion (CAP: Computer Aided Planning), sondern es werden in immer stärkerem Maße auch die Leitungs-, Planungs- und Verwaltungsprozesse, Qualitätssicherung (CAQ: Computer Aided Quality Assurance), Transport, Lagerhaltung, Absatz usw. integriert. Das Ziel besteht in einer durchgängigen Automatisierung aller betrieblichen Prozesse, deren Realisierung durch eine hochautomatisierte Fabrik erfolgt. Diese Aufgabe wird durch den Begriff CIM — Computer Integrated Manufacturing — erklärt [3] und [4].

Ein Schlüsselproblem zur Lösung dieser Aufgaben stellt die Realisierung der Kommunikationsanforderungen innerhalb eines CIM-Systems dar. Neben der bisherigen Mensch-Maschine-Kommunikation tritt als qualitativ neues Element im hochautomatisierten Produktionsbetrieb die Kommunikation zwischen Computern bzw. computerisierten Maschinen und Anlagen sowie zwischen CAD/CAM-Systemen und deren Leitungsinformationssystem hinzu. Dies erfordert eine Verkopplung der dezentralen Einrichtungen im betrieblichen Nah- und Fernbereich, die zusätzlich auch für eine Mensch-Mensch-Kommunikation nutzbar ist. Die Informationsübermittlung zwischen den Systemen erfolgt über bereits existierende oder über neue Kommunikationseinrichtungen [6] und [7].

An die Kommunikationseinrichtungen für einen hochautomatisierten Produktionsprozeß (CIM) werden folgende Anforderungen gerichtet [6]:

- Datentransfer zwischen CAD- und CAM-Stationen, im Qualitätssicherungssystem (CAQ) und im Leitungsinformationssystem (MIS)
- Programmtransfer, z. B. flexibles Programmladen eines Fertigungssystems von einer entfernten Programmbank
- Bürokommunikation, z. B. Mailbox-Dienst für innerbetriebliche Nachrichten
- Transfer von Dokumenten, technischen Zeichnungen
- Bedienerkommunikation
- Austausch von Steuerungsinformationen zwischen dezentralen Fertigungsprozessen
- verteilte Verarbeitung.

1. Lokale Rechnernetze

Zur Realisierung der innerbetrieblichen Kommunikationsprozesse werden im automatisierten Betrieb vorrangig lokale Rechnernetze (LAN: Local Area Network) eingesetzt. Sie eignen sich insbesondere für die neuen Datenkommunikationsdienste, sowohl in der Bürokommunikation, im Leitungsinformationssystem als auch im rechnergestützten Produktionssteuerungssystem. Lokale Rechnernetze haben i. allg. eine ring- oder linienförmige Struktur. Als Architekturmodell dient das OSI-Referenzmodell der ISO (ISO 7498). Für CIM-typische Einsatzfälle wurden international zwei Kommunikationsmodelle entwickelt, die beide auf den ISO-Standard-Kommunikationsprotokollen beruhen:

1. MAP (Manufacturing Automation Protocol):
   Es stellt Dienste für die flexible Fertigungssteuerung bereit; es basiert auf der determinierten Token-Bus-Zugriffstechnik unter Nutzung von Koaxialkabel mit Breitbandübertragung.
2. TOP (Technical Office Protocol):
   Es stellt Dienste für den Austausch technischer Daten (z. B. in CAD-Systemen) und für das Leitungs- und Planungssystem bereit; es basiert auf der stochastischen CSMA/CD-Zugriffstechnik und Basisbandübertragung.

Die Anforderungen an das Kommunikationssystem werden durch die zu übertragenden Datenmengen und zu realisierenden Antwortzeiten (im Bereich der Produktionszellen sogar mit Echtzeitanforderungen) gestellt. Die Größe der Nachrichten bzw. Files liegt zwischen 1 und 500 Kbyte, bei grafischen Daten (CAD) im Bereich von 1 ... 10 Mbyte. Für die Antwortzeiten gelten Werte im Bereich 0,5 ... 10 s, im Produktionsprozeß 0,01 ... 0,5 s. Daraus resultieren Durchsatzanforderungen von 0,1 ... 10 Mbit/s. International wird auf 10 Mbit/s-LAN orientiert. Für die bürokommunikationstypischen Übertragungsvorgänge nutzt man CSMA/CD-LAN (z. B. Ethernet), für produktionsorientierte Übertragungsvorgänge mit Echtzeitbedingungen werden Tokenbus- und -ringnetze eingesetzt.

Die Einordnung von MAP bzw. TOP in die OSI-Architektur zeigt die Tafel.

2. Leistungsbewertung des Kommunikationssystems

Die Forderung nach effizientem und funktionssicherem Einsatz der zu installierenden verteilten Rechnersysteme als Basis des Kommunikationssystems bedingt die Untersuchung des Systemverhaltens, um Leistungskennziffern und Grenzen der Leistungsparameter ermitteln zu können. Diese Aufgabe ist zusätzlich dadurch gekennzeichnet, daß die Komplexität und der Kompliziertheitsgrad der zu analysierenden Systeme sehr hoch und differenziert ist sowie Werkzeuge zur Leistungsbewertung noch nicht im allgemeingültigen Maße zur Verfügung stehen.
Grundlage der Leistungsbewertung ist die Modellierung in ihrer Einheit von Modellbildung, Analyse/Synthese und Bewertung. Durch Abstraktion und Transformation werden für Rechnersysteme Verhaltensmodelle erstellt, die entsprechend der verwendeten Methodik analytischen, simulativen, empirischen oder hybriden Charakters sind.
Ein wichtiges Kriterium zur Auswahl des entsprechenden Verhaltensmodells stellt neben der Beschreib- und Lösbarkeit das Vorhandensein eines ingenieurtechnisch einsetzbaren Werkzeuges (Tool) dar. Im internationalen Rahmen existieren auf diesem Gebiet einige rechnergestützte Verfahren zur Leistungsbewertung von Rechner- und Rechnerverbundsystemen. In den letzten Jahren wurden hierzu auch in der DDR Modellierungsmethodiken und -werkzeuge entwickelt und eingesetzt.
Grundlage der Leistungsbewertung des Kommunikationssystems stellt das vom Autor entwickelte Analysewerkzeug BNETD dar, das durch Modellierungsbausteine ergänzt ist. In der Grundkonzeption erfolgt die Modellierung auf der Basis analytischer Verhaltensmodelle, wobei insbesondere bedienungstheoretische Verfahren (Bedienungssysteme, Bedienungsnetze) zur Anwendung kommen. BNETD und zugehörige Modellierungswerkzeuge sind sowohl für ESER-Rechner (OS/ES) als auch für Büro- und Personalcomputer (SCP bzw. DCP) implementiert [1] und [2].
Die Leistungsbewertung des Kommunikationssystems auf Basis lokaler Rechnernetze (OSI-Architektur, unterschiedliche Protokolle und Zugriffsverfahren) kann sowohl separat als auch integriert im Gesamtsystem erfolgen. Zur Unterstützung der Analyse des Systemverhaltens kommunikationsorientierter Systeme wurde das Analysesystem LANEX entwickelt. Es existiert in 2 Versionen: für den Arbeitsplatzcomputer A 7100 als eigenständiges Programm LANEX (SCP 1700, TURBO-PASCAL 3.0) bzw. für einen kompatiblen 16-bit-Personalcomputer unter MS-DOS als Modellklasse LAN innerhalb des allgemeinen Modellierungssystems DIMPES (TURBO-PASCAL 4.0).
Unter Verwendung verschiedener Modellierungsmethoden (Dekompositionsmethoden, Top-down- und Bottom-up-Strategien, Mehrebenen-Konzept) können die datenübertragungsspezifischen, protokollorientierten und auch die verarbeitungsspezifischen Vorgänge (z. B. Bürokommunikation, Filetransfer, verteilte Verarbeitung) analysiert und bewertet werden. Die Validierung des Modells erfolgte durch Simulationsexperimente und Systemmessungen. Die Parametrisierung des Modells ist vom Anwender über Erwartungswerte (Schätzung, Messung) vorzunehmen. Für Standardnetze existieren Voreinstellungen. Die Erprobung des Analysewerkzeuges wurde an realen Netzkonfigurierungen durchgeführt.

Tafel. Einordnung von MAP und TOP in die OSI-Architektur

Schicht	TOP	MAP
7	ISO FTAM (DP) 8571 • File Transfer Protocol • File Management (limited)	ISO FTAM (DP) 8571 • File Transfer Protocol • Manufacturing Messaging Format Standard (MMFS) • Common Application Service Elements (CASE)
6	–	–
5	ISO Session Protocol (IS) 8327	
4	ISO Transport Protocol (DIS) 8073, class 4 (datagram service)	
3	ISO Internetwork Protocol (DIS) 8473, connectionless and for X.25	
2	ISO Logical Link Control (DIS) 8802/3, type 1, class 1 (IEEE 802.2)	(connectionless)
1	ISO CSMA/CD (DIS) 8802/3 (IEEE 802.3) CSMA/CD medium access control (stochastic, office)	ISO Token-passing bus (DIS) 8802/4 (IEEE 802.4) Token-passing bus medium access control (deterministic, manufacturing)

Literatur

[1] IRMSCHER, K.: Programmsystem BNETD zur Analyse geschlossener Bedienungsnetze. Wiss. Zeitschrift der TU Dresden 34 (1985) 4, S. 87–94.
[2] IRMSCHER, K.: Rechnergestützte Modellierung und Bewertung. edv-aspekte (1986) 4, S. 9–11.
[3] KLOSE, J.; KOCHAN, D.: Durchgängige automatisierte Systeme. edv-aspekte (1985) 4, S. 2.
[4] KOZIOLEK, H.; SALECKER, W.; SCHMIEDER, TH. (Hrsg.): CAD/CAM. Leitungsaufgaben und Erfahrungen. Berlin: Verlag Die Wirtschaft 1987.
[5] LÖFFLER, H.: Lokale Netze. Berlin: Akademie-Verlag 1987.
[6] LÖFFLER, H.: Kommunikationsmodell und -technologie des künftigen Maschinenbaubetriebes auf der Basis lokaler Rechnernetze. Kolloquium „Informations-, Kommunikations- und Automatisierungsmodell des rechnerintegrierten Maschinenbaubetriebes im Sozialismus". TU Dresden, 15. April 1987.
[7] NEUBERT, P.: Merkmale eines Kommunikationssystems der Zukunft – Entwicklungstendenzen in der Nachrichten- und Rechentechnik. Wiss. Zeitschrift der TU Dresden 36 (1987) 3, S. 235–240.

msr 8830

Doz. Dr. sc. techn. K. IRMSCHER, TU Dresden, Informatik-Zentrum, WB Rechnersysteme, Mommsenstr. 13, Dresden, 8027

(Fortsetzung von S. 257)

[19] RUDOLPH, K.; STANEK, W.: Entwicklung eines Modellsystems zur dynamischen Projektierung integrierter automatisierter Fertigungen. Dissertation B, TH Karl-Marx-Stadt 1982.
[20] SCHMIGALLA, H.: Zur Notwendigkeit und Möglichkeit der Simulation in der Betriebsgestaltung. Forschungsergebnisse der FSU Jena, Nr. 55 (N), 1978.
[21] GRUNDIG, C.-G.: Fertigungsprozeßsteuerung unter Ausnutzung von Reihenfolge- und Bestandseffekt. Fertigungstechnik und Betrieb, Berlin 31 (1981) 3, S. 157–159.
[22] SCHMIGALLA, H.: Ein Mengen-Zeit-Modell zur analytischen Bestimmung von Grundgrößen des Produktionsverlaufes. Vortrag auf der 4. Fachtagung der KDT „Moderne Fertigungsorganisation und Fertigungssteuerung", März 1985, Dresden.
[23] NALIMOV, V. V.: Theorie des Experimentes. Berlin: VEB Deutscher Landwirtschaftsverlag 1975.
[24] KAPLICK, K.; LORENZ, G.: Experimentelle Verfahren zur Prozeßanalyse. Leipzig: VEB Deutscher Verlag für Grundstoffindustrie 1973.
[25] JAHN, W.; KÜHNE, P.; VAHLE, H.: Analyse und statistische Prozeßmodellierung für die Prozeßsteuerung. Bd. 128 der REIHE AUTOMATISIERUNGSTECHNIK. Berlin: VEB Verlag Technik 1972.
[26] ERKMANN, J.; GRUNDIG, C.-G.: Nutzung des rechnergestützten Projektierungssystems RPS-FAIF zur Gestaltung eines integrierten gegenstandsspezialisierten Fertigungsabschnittes. Fertigungstechnik und Betrieb, Berlin 36 (1986) 7, S. 407–411.
[27] WIRTH, S.; ERKMANN, J.; FÖRSTER, A.: CAD-Projektierungssystem für flexible Fertigungssysteme und Fertigungsstätten. Fertigungstechnik und Betrieb, Berlin 36 (1986) 9, S. 542–545.

msr 8858

R. Wieland

LISP — eine Einführung für Automatisierungstechniker 6

2.4.4. String/Char

Wie in fast allen Programmiersprachen existieren auch in LISP die Datentypen String und Char. Über diesen Datentypen gibt es eine Menge von Funktionen, die sich aber nicht wesentlich von den Möglichkeiten in anderen Programmiersprachen unterscheiden. Aus diesem Grund und aus Platzgründen werden hier nur ein paar besonders wichtige genannt, ansonsten sei aber auf die Anwenderdokumentation der entsprechenden LISP-Implementierung verwiesen.
Auf die Darstellung der Funktionen auf dem Datentyp Char, die im wesentlichen Konvertierungen (Umwandlungen von Klein- in Großbuchstaben, von ASCII in Hexadezimalzahlen usw.) umfassen, wird hier völlig verzichtet. Diese Funktionen bieten keine Besonderheiten und werden nur bei der Behandlung relativ maschinennaher Problemstellungen genutzt. Verglichen mit diesen doch recht speziellen Anwendungen kommt den „Stringfunktionen" doch eine größere Bedeutung zu. Eine große Rolle spielen diese Funktionen auch in den später zu erläuternden Funktionen zur Ein-/Ausgabe, die häufig als Datentyp Strings erwarten.
Die Verknüpfung zweier Strings ist mit STRING-APPEND möglich:

(STRING-APPEND string1 string2) ⇒ string1 + string2
(STRING-APPEND "C:\\" "USER") ⇒ "C:\\ USER"

Eine derartige Verknüpfung ist z. B. bei Zugriffen auf externe Speicher häufig (LISP benötigt \\ statt \ für den Übergang zu Unterdirectories in MS-DOS).
Einen Vergleich zweiter Strings erlauben:

(STRING= string 1 string 2 &KEY start1 end1 start2 end2) ⇒ bool
(STRING< string 1 string 2 &KEY start1 end1 start2 end2) ⇒ bool
(STRING-EQUAL string1 string2 &KEY start1 end1 start2 end2) ⇒ bool
(STRING-LESSP string1 string2 &KEY start1 end1 start2 end2) ⇒ bool

STRING-EQUAL und STRING-LESSP ignorieren den Unterschied, der aus der Verwendung von Groß- und Kleinschreibung stammt.
Zur Suche von Substrings gibt es folgende Funktionen:

(STRING-SEARCH key string &OPTIONAL from to) ⇒ NIL/Index
(STRING-SEARCH* key string &OPTIONAL from to) ⇒ NIL/Index

Mit diesen Funktionen kann nach einem key (Teilstring) in einem String gesucht werden. Die Angabe des Suchbeginns (from) und des Suchendes (to) ist optional. Zurückgegeben wird die Stelle des Anfangs des Substrings (Index) bzw., wenn der Substring nicht im String enthalten war, NIL. STRING-SEARCH* ignoriert wieder Unterschiede aus der Groß- und Kleinschreibung.
Zur Teilung von Strings dienen die Funktionen:

(STRING-LEFT-TRIM charakter-bag string) ⇒ Trimmed-String
(STRING-RIGTH-TRIM charakter-bag string) ⇒ Trimmed-String

Dabei müssen beide Argumente vom Typ String sein.
Eine besonders wichtige Typkonvertierung ist die Umwandlung eines Strings in ein Symbol. Diese Umwandlung ist dort notwendig, wo über die noch zu behandelnden E/A-Funktionen eingegebene Strings als interne Symbole repräsentiert werden sollen. Diese Symbole können dann beispielsweise als Namen von Listen Verwendung finden. Für derartige Aufgaben existiert die Funktion:

(READ-FROM-STRING string) ⇒ Symbol

Umgekehrt wandelt die Funktion

(STRING symbol) ⇒ string

ein Symbol in einen String.

2.5. Eingabe-/Ausgabefunktionen

In den bisherigen Beispielen wurde der Read-eval-print-Zyklus des LISP-Interpreters für die Ein- und Ausgabe genutzt. Dieser doch etwas enge Rahmen soll nun durch die Einführung spezieller Funktionen zur Ein- und Ausgabe gesprengt werden. LISP bietet dazu vielfältigste Funktionen, angefangen von der zeichenweisen Ein- und Ausgabe über das Lesen und Schreiben als primitive Funktion bis hin zur universellen formatierten Anweisung. Hier widerspiegeln sich auch zahlreiche Bezüge zum Betriebssystem MS-DOS. Die Liste der mit den Eingabe-/Ausgabefunktionen in Zusammenhang stehenden Befehle ist derart umfangreich, daß auch hier wieder ausgewählt werden mußte. Begonnen wird mit der zeichenweisen Ein- und Ausgabe, die vor allem bei Nutzung externer Speicher Verwendung findet.

(WRITE-CHAR char &OPTIONAL stream) ⇒ char
(READ-CHAR &OPTIONAL stream) ⇒ char
(WRITE-BYTE integer &OPTIONAL stream) ⇒ integer
(READ-BYTE &OPTIONAL stream) ⇒ integer

Mit diesen Funktionen können Zeichen byteweise geschrieben bzw. gelesen werden. Der Stream gibt die aktuelle Datei an. Über das Stream-Konzept wird noch ausführlicher zu berichten sein. Ohne gesonderte Vereinbarung dienen als aktuelle Lesedatei das Keyboard und als aktuelle Schreibdatei das Terminal. Die Zahlenbasis der Integer-Größe beträgt standardmäßig 10, kann aber durch Belegung der Variablen *READ-BASE* bzw. *PRINT-BASE* im Bereich von 2 bis 36 eingestellt werden. Die zeichenweise Ein- und Ausgabe spielt vor allem bei der Dateiarbeit eine große Rolle, für die Ein- und Ausgabe auf den Standardgeräten gibt es in LISP leistungsfähigere Funktionen.

(READ)	Symbol wird über die Tastatur eingelesen (Abschluß ET)
(READ-LINE)	Einlesen einer Zeile
(PRINT obj)	Ausgabe des obj mit nachfolgendem nl
(PRIN1 obj)	Ausgabe des obj ohne nl
(TERPRI)	nl

Diese Funktionen, obwohl vorwiegend für die Bildschirmausgabe gedacht, können natürlich im Zusammenhang mit anderen Dateien und Ausgabegeräten genutzt werden. Mit ihnen ist schon eine recht ordentliche Ausgabe möglich, die für viele Testzwecke ausreicht. Eine wirklich universelle Ausgabefunktion steht in LISP aber mit

(FORMAT destination control-string &REST arguments)

zur Verfügung. Darin bedeuten:

destination	Ausgabe-stream (T = BS)
control-string	erlaubt die Formatierung mit folgenden Steuerzeichen (Auswahl)
~A	ASCII
~D	dezimal
~O	oktal
~H	hexadezimal

~B binär
~% nl

Als Beispiel soll folgende Sequenz auf dem Bildschirm ausgegeben werden:

nl dezimal: 255 hexadezimal: FF nl

Das kann geschehen mit:

```
(FORMAT T " ~% dezimal: ~D hexadezimal: ~H ~%"
          255 255)
```

Die erste 255 wird durch D dezimal ausgegeben, die zweite 255 wird hexadezimal konvertiert und ausgegeben.

Zur Realisierung der Zugriffe auf externe Speichermedien, wie Disketten und Festplatten, ist es notwendig, die zu bearbeitenden Dateien zu öffnen und nach Abschluß des Zugriffs wieder zu schließen. Dazu dienen die Funktionen OPEN und CLOSE.

(OPEN pathname &KEY direction element-type ⇒ stream

(CLOSE pathname)

Hierbei bedeuten:

pathname	Name einer Datei in MS-DOS Konvention
direction	input / output
element-type	string-char / unsigned-byte
stream	Funktionsobjekt

In LISP werden Streams als Funktionsobjekte (sog. Closures) verwaltet. Es ist möglich, auf diese Streams auch Operationen anzuwenden, z. B. die Positionierung eines Zeigers. Damit wird es beispielsweise möglich, den Direktzugriff zu organisieren. Eine detailliertere Behandlung der Closures erfolgt in einem späteren Abschnitt. An dieser Stelle werden nur die Möglichkeiten der Behandlung von Closures aufgezeigt, soweit sie für die Dateiarbeit von Interesse sind. Um die Dateiarbeit zu demonstrieren, sollen zuerst zwei Funktionen angegeben werden, die ein sequentielles Lesen von einer und Schreiben auf eine Datei ermöglichen. Bevor diese Funktionen angegeben werden können, müssen noch zwei Hilfsfunktionen eingeführt werden. Mittels der in LISP implementierten Funktion MULTIPLE-VALUE-BIND ist es möglich, in einem Schritt gleich mehrere Werte zu binden. Dieses wird bei der File-Arbeit benötigt, um die File-Informationen einer MS-DOS-Datei, wie die Attribute, die Größe, das Erstellungsdatum und die Uhrzeit, zu binden. Die File-Informationen liefert die Funktion FILE-INFO. Damit ist man in der Lage, eine Funktion get-file, die ein spezifiziertes File liest und als Feld (String) zurückgibt, zu implementieren. (Hierbei ist zu beachten, daß Strings intern als Felder abgespeichert werden.)

```
(DEFUN get-file (pathname)
  (MULTIPLE-VALUE-BIND (attr sz-hi sz-lo date time)
    (FILE-INFO pathname)
    (SETF get-stream (OPEN pathname
                           :element-type 'unsignend-byte)
          file      (MAKE-ARRAY sz-lo))
    (DOTIMES (i sz-lo)
      (SETF (AREF file i) (READ-CHAR get-stream)))
    (CLOSE get-stream)
    file)
```

Erklärung:

Nachdem mit MULTIPLE-VALUE-BIND und FILE-INFO die entsprechenden Parameter (attr sz-hi sz-lo date time) gebunden wurden, werden mit SETF ein Stream eröffnet und global an die Variable get-stream gebunden. Weiterhin wird ein globaler Vektor „file“ entsprechend der Filegröße (hier vereinfachend nur für kurze Files geeignet, da nur sz-lo genutzt wurde) eröffnet. Dieser Vektor nimmt dann den zu lesenden String auf. Dazu dient die DOTIMES Schleife. Abschließend wird der eröffnete Stream mittels CLOSE geschlossen. Die Funktion getfile gibt dann den Vektor „file“ zurück.

(wird fortgesetzt)

F. Roß

PROLOG — eine Einführung für Automatisierungstechniker 6

Im folgenden werden die mathematischen Operatoren beschrieben, die in PROLOG gemäß der Edinburgh-Syntax umgesetzt sind. Erweiterungen oder Änderungen für die einzelnen PROLOG-Systeme müssen im entsprechenden Handbuch nachgeschlagen werden. Die mathematischen Operatoren können in Vergleichsoperatoren und Arithmetikoperatoren eingeteilt werden. Standardmäßig sind sie Infixoperatoren. Als Vergleichsoperatoren stehen zur Verfügung:

=	gleich
\=	ungleich, nicht in jedem PROLOG-System implementiert
<	kleiner
>	größer
=<	kleiner gleich, nicht <=
>=	größer gleich, nicht => .

Außer bei Test auf Gleichheit bzw. Ungleichheit (siehe auch Abschnitt 4.) müssen eventuell vorhandene Variablen als Operanden mit einem Wert belegt sein, ansonsten wird das Programm mit einer Fehlermeldung abgebrochen.

Beispiele:

```
?—4<5.
Yes
?—X=5,X\=2.
X=5
Yes
?—X=5,X\=Y.
No
```

Die Systemantwort im letzten Beispiel ist klar, da gemäß Abschnitt 4. die beiden Variablen in der zweiten Teilaussage gleich sind (X ist mit 5 instanziert, Y ist ungebunden) und damit nicht ungleich sein können.

Folgende Arithmetikoperatoren werden vom PROLOG-System zur Verfügung gestellt:

+	Summation
—	Subtraktion
*	Multiplikation
/	Division
mod	modulo.

Eventuell vorhandene Variablen als Operatoren müssen grundsätzlich instanziert sein. Geschachtelte arithmetische Operationen sind erlaubt. Die Priorität in der Abarbeitung ist standardmäßig wie folgt festgelegt:

mod höchste Priorität
*, /
+,— niedrigste Priorität.

Die mathematische Wertzuweisung erfolgt mit dem Operator „is“.

Beispiele:

```
?—X is 5*2+1.
X=11
Yes
```

?—X is 5*(2+1).
X=15
Yes
?—6 is 4+2.
Yes.

Operatoren können in PROLOG auch selbst definiert werden. Dazu dient das vordefinierte Prädikat (eingebautes Prädikat) „op(X,Y,Z)“. X ist die Präzedenzklasse und legt die Priorität des Operators fest. Der Wert kann zwischen 1 und 255 liegen, wobei der Wert 1 der höchsten Priorität entspricht. Y gibt die Position und die Assoziativität des Operators an. Die Positionsangabe kann lauten:

fx,fy	Präfixoperator
xf,yf	Postfixoperator
xfy,yfx,xfx,yfy	Infixoperator.

Die Assoziativität gibt an, wie Operatoren gleicher Priorität abgearbeitet werden. Assoziativitätsangaben können sein:

xfy	rechtsassoziativ
yfx	linksassoziativ
xfx,yfy	reflexiv.

Z gibt den Namen des Operators an.
Als Beispiel werden die Prioritäten der Multiplikation und der Summation auf verschiedene Werte gesetzt und deren Einfluß auf das Ergebnis betrachtet:

?—op (20,xfy, '+').
?—op (10,xfy, '*').
?—X is 4*5+1.
X=21
Yes

Mit den ersten beiden Klauseln wurde der Multiplikation eine höhere Priorität zugewiesen als der Summation. Sie wurde auch zuerst ausgeführt.

?—op (20,xfy, '*').
?—op (10,xfy, '+').
?—X is 4*5+1.
X=24
Yes

Hier wurde die Priorität für die Summation höher festgelegt, und somit wurde diese Operation auch zuerst ausgeführt.

7. Eingebaute Prädikate

Wie schon mehrfach angedeutet, stellt ein PROLOG-System eine Anzahl vordefinierter Prädikate dar. Da diese Prädikate fest in das System integriert sind, heißen sie eingebaute Prädikate. Sie realisieren hauptsächlich Ein- und Ausgabe, Dateiarbeit, Termklassifikation, Klauselmanipulation und Programmtracing (Verfolgen des Programmablaufs im Schrittbetrieb). Die Anzahl der eingebauten Prädikate unterscheidet sich sehr stark von System zu System. Die Art der Prädikate hängt stark vom Rechnertyp ab, auf dem das PROLOG-System implementiert wurde. Zu Beginn der Arbeit mit einem konkreten System sollte man sich deshalb anhand des Handbuches genau über die eingebauten Prädikate informieren, um die Möglichkeiten des Systems voll ausschöpfen zu können.

8. Sprachkopplung

Im Abschnitt 6. wurde schon darauf hingewiesen, daß die Programmiersprache PROLOG nicht für die Lösung aller Probleme geeignet ist. Mathematische Algorithmen sollten z. B. in herkömmlichen prozeduralen Sprachen umgesetzt werden. Ist für die Lösung einer Aufgabe aber sowohl symbolische als auch numerische Verarbeitung notwendig, ist die Wahl der Programmiersprache besonders schwierig. Eine Möglichkeit wäre die Wahl einer prozeduralen Sprache, wobei für die symbolische Verarbeitung ein Inferenzmechanismus ähnlich dem eines PROLOG-Systems geschrieben werden muß. Eine andere und wohl die einfachere und elegantere Lösung ist die Kopplung von PROLOG mit einer oder mehreren prozeduralen Programmiersprachen [23]. Grundsätzlich gibt es drei Möglichkeiten der Kopplung von PROLOG mit anderen Programmiersprachen:

a) Programme, die in unterschiedlichen Sprachen geschrieben wurden, werden nacheinander vom Nutzer gestartet und greifen auf eine gemeinsame Wissensbasis in Form einer Datei zu. Diese wohl einfachste Lösung wird vor allem dann angewendet, wenn nur selten, meist für Auswertezwecke, sowohl numerische als auch symbolische Verarbeitung notwendig ist. Diese Methode ist prinzipiell mit allen PROLOG-Systemen realisierbar. Häufigstes Anwendungsgebiet ist die Nutzung von Datenbanken durch ein PROLOG-System.
b) Ein vom Nutzer gestartetes Programm ruft während der Abarbeitung andere Programme auf. Die Datenübergabe erfolgt über externe Dateien oder über den direkt adressierten Speicher. Diese Methode ist stark vom verwendeten Betriebssystem abhängig. Sie kommt vor allem in der Programmerstellungsphase zum Einsatz und wenn fertige, z. B. kommerzielle Programme genutzt werden sollen. PROLOG-Systeme, die diese Kopplungsmethode unterstützen, sind z. B. C-PROLOG und TURBO-PROLOG. Auch Rahmenexpertensysteme verwenden diese Methode häufig.
c) Die Programme werden in einen Zwischenkode übersetzt und anschließend zu einem einzigen Programm verbunden. Diese Methode ist in der Abarbeitung die schnellste, da die Datenübergabe direkt über die Prozeduraufrufe erfolgt, jedoch ist sie stark vom verwendeten PROLOG-System und dem Compiler der verwendeten prozeduralen Sprache abhängig. Ein PROLOG-System, das diese Methode unterstützt, ist TURBO-PROLOG.

Keiner der beschriebenen Methoden kann der Vorzug gegeben werden. Die Wahl des konkret zu verwendenden Kopplungsverfahrens hängt vielmehr vom zu lösenden Problem ab. Deshalb sollte vor dem Programmieren eine klare Strukturierung des Wissens erfolgen und damit auch die Art der Wissensverarbeitung geklärt werden.

9. micro-PROLOG

Zum Abschluß soll noch ein PROLOG-System kurz vorgestellt werden, das eine andere Syntax als das „Edinburgh-PROLOG“ hat, aber auf 8-bit- und 16-bit-PC weit verbreitet ist, das micro-PROLOG. Die Abarbeitung eines micro-PROLOG-Programmes ist identisch mit der eines Standard-PROLOG-Programms. Es existieren aber grundsätzliche syntaktische Unterschiede. Auf diese Unterschiede soll im folgenden kurz eingegangen werden.
Ein Atom kann auch mit einem Großbuchstaben beginnen. Als Zahlenkonstanten sind Realzahlen möglich. (Die eingebauten Prädikate beginnen z. B. alle mit einem Großbuchstaben). Als Variablen gelten die Buchstaben X, Y, Z, x, y und z. Diesen können Zahlen von 1 bis 26 folgen.

Beispiele:

X
Y
Z15

Listen werden durch runde Klammern begrenzt. Ihre Elemente werden nicht durch Kommata, sondern durch Leerzeichen getrennt.

Beispiel:

() leere Liste
(Anlage_1 2 4)

Strukturen entsprechen exakt einer Liste, wobei das erste Element als Prädikat betrachtet wird.

Beispiel:

(Zustand Anlage_1 frei)

(wird fortgesetzt)

Jahresversammlung 1988 des Fachausschusses Metrologie und Standardisierung der WGMA

Der Fachausschuß Metrologie und Standardisierung der WGMA hat seine Jahresversammlung 1988 zum Thema „Künstliche Intelligenz und ihre Nutzung in der Meßtechnik" durchgeführt. Die in den 50er Jahren in den USA geprägte Wortbildung „artificial·intelligence", die wir mit „künstliche Intelligenz" (KI) übersetzen, ist schnell zu einem Schlagwort geworden. Ohne daß es bereits eine allgemein akzeptierte Definition der KI gibt, wird weltweit zum Gegenstand KI geforscht. Es mag legitim sein, den Terminus KI für einen unscharfen Begriff zu benutzen und ihm mit fortschreitender Erkenntnis schrittweise einen wissenschaftlichen Inhalt zu geben. Gleichzeitig aber ist es notwendig, eine Abgrenzung vorzunehmen und sich darüber zu verständigen, was alles nicht zur KI gezählt werden soll. Beides wird gegenwärtig getan, wobei sich in einzelnen Fragen konträre Auffassungen gegenüberstehen. Künstliche Intelligenz wird heute vor allem durch Expertensysteme (ES) repräsentiert, deren vorrangige Aufgabe in der von Mensch und Computer gemeinsamen Lösung logischer Probleme besteht. Darüber hinaus werden in den Technikwissenschaften Termini benutzt, die von einer KI abgeleitet sind. So spricht man von intelligenten Robotern, intelligenten Sensoren und auch von intelligenter Meßtechnik.

Der Fachausschuß Metrologie und Standardisierung der WGMA hat es sich zur Aufgabe gemacht, einen Beitrag zur Entwicklung der metrologischen Grundlagen und deren praktischer Anwendung in der Meßtechnik zu leisten. Zu den Grundlagen eines Wissenschaftszweiges gehört auch die Terminologie. Eine seit längerem im Fachausschuß geführte Diskussion war darauf gerichtet, das „Für" und „Wider" der Verwendung des Terminus „intelligente Meßtechnik" in der wissenschaftlichen Arbeit zu erwägen. Der Fachausschuß will sich ausdrücklich nicht dort äußern, wo es um Werbung und um kommerzielle Fragen geht.

Die Diskussion zur Jahresversammlung 1988 wurde mit zwei wissenschaftlichen Vorträgen eingeleitet. H. Liebscher (Humboldt-Universität zu Berlin) sprach zu einigen philosophischen Aspekten der KI; W. Weller (Humboldt-Universität zu Berlin) erläuterte in seinem Vortrag, was gegenwärtig unter „lernfähigen Systemen" zu verstehen ist und wo die Schwerpunkte der weiteren Forschungsarbeit auf diesem Gebiet liegen. In beiden Vorträgen wurde deutlich, daß die Forschung zur KI ganz am Anfang steht und es einer weiteren wesentlichen Leistungssteigerung von Computern, Speichern und der Software bedarf, um nennenswerte Fortschritte zu erzielen. Hierauf ist auch an anderer Stelle von kompetenten Fachleuten verwiesen worden. Die Diskussion im Fachausschuß ging davon aus, daß mit Hilfe von Rechnern die Leistungsfähigkeit der Meß- und Informationstechnik wesentlich erhöht werden konnte. Der Einsatz von Rechnern und spezieller Software erlaubt eine Erweiterung der gerätetechnisch realisierbaren Funktionen und metrologischen Eigenschaften eines Meßmittels und gestattet eine umfassendere Meßsignalverarbeitung, die zu zusätzlichen Informationen führt. Qualitative Erweiterungen eines Meßmittels werden erreicht durch die Fähigkeit zur Selbstkalibrierung und Selbstdiagnose, die Veränderung von Kennlinien, die Anpassung an unterschiedliche innere Bedingungen und äußere Einflüsse, die Korrektur systematischer Fehler, die Reproduktion der Meßbereitschaft, die Steuerung des Meßzyklus u. a. Aus den Meßwerten lassen sich mit Hilfe des Rechners ausgezeichnete Werte (Mittelwert, Grenzwerte u. a.) bestimmen, Standardabweichungen und Trends berechnen, Soll-Ist-Vergleiche durchführen u. a. Solche qualitativen Erweiterungen und quantitativen Mehrleistungen der Funktionen eines Meßmittels bzw. des meßwertverarbeitenden Systems sind eindeutig determiniert und laufen nach vorgegebenen, zum Teil relativ einfachen Algorithmen ab. Mehrheitlich vertreten die Mitglieder des Fachausschusses die Auffassung, daß es sich hierbei um keine Intelligenzleistungen des Meßmittels oder des Meßverfahrens handelt. Insbesondere wird ein adaptives Verhalten technischer Systeme nicht zur KI gezählt. Eine solche Auffassung kommt auch in dem Fachausschuß zugegangenen schriftlichen Äußerungen namhafter Wissenschaftler des Fachgebietes zum Ausdruck.

Es wäre denkbar, daß ein Rechner künftiger Generationen mit sehr viel gespeicherter „Erfahrung" die Abarbeitung eines heuristischen Programms selbständig übernehmen könnte und auch gewisse auf Erfahrung beruhende Entscheidungen trifft oder selbständig gewisse Plausibilitätstests, wie sie in der Meßtechnik üblich sind, vornimmt. Auch wenn nicht ausgeschlossen wird, daß sich das System Rechner-Meßmittel ohne A-priori-Voraussetzungen für die Lösung einer komplizierten Meßaufgabe selbst organisiert, bedarf es noch sehr intensiver Forschungs- und Entwicklungsarbeit, bis ein solcher Zustand erreicht sein wird.

Die Mitglieder des Fachausschusses, die sich bisher in der Diskussion geäußert haben, kommen mehrheitlich zu der Auffassung, daß die Grundlagen der Meßtechnik vorrangig dem gesicherten Wissen der Physik entstammen und die Meßtechnik quantitative Informationen in Form von Meßdaten zu liefern hat. Eine informationstechnische Aufbereitung und Weiterverarbeitung der Meßdaten mit dem Ziel, logische Entscheidungen zu treffen, kann nicht die Aufgabe der Meßtechnik sein. Damit wäre sie überfordert. Deshalb sollte man beim gegenwärtigen Stande der KI-Forschung nicht so tun, als gäbe es eine „intelligente Meßtechnik", über die man bereits verfügen könne. Es sei denn, unter „Meßtechnik" bzw. „messen" wird etwas anderes verstanden, als es in Wörterbüchern der Metrologie zu lesen und in der DDR in gültigen Standards festgelegt ist. Der Fachausschuß kann deshalb die Verwendung der Wortbildung „intelligente Meßtechnik" als metrologischen Begriff in der wissenschaftlichen Arbeit und in der Aus- und Weiterbildung gegenwärtig nicht empfehlen.

msr 9031 Dr. rer. nat. Karl Schmidt, Vorsitzender des FA Metrologie und Standardisierung

Die Auffassung der Redaktion msr zur intelligenten Meßtechnik weicht in einigen Punkten vom hier dargelegten Standpunkt ab. Dem aufmerksamen Leser wird nicht entgangen sein, daß auch in der msr einige Beiträge zu dieser Thematik erschienen sind. Akzeptiert man z. B. die Formulierung aus msr, Berlin 32 (1989) 3, S. 98:

„Die hohe Leistungsfähigkeit des Menschen bei der gezielten Beeinflussung seiner Umgebung basiert auf dem Zusammenspiel umfangreicher Sensorik, vielfacher Freiheitsgrade der Aktorik/Motorik und einer geeigneten, hierarchisch organisierten Informationsverarbeitung, die als Intelligenz bezeichnet werden kann. „Intelligenz" ist dabei als Maß dafür anzusehen, wie geschickt Information abgespeichert und dann ziel- oder verhaltensorientiert verarbeitet bzw. zu neuer, i. allg. kompakterer Information synthetisiert wird.",

so erscheint der Begriff „intelligente Meßtechnik" angesichts zahlreicher, auch industrieller Sensorsysteme, die Datenaufbereitung, -vorverarbeitung, -verdichtung bzw. -auswertung und -korrektur, z. B. in Robotersteuerungen, mit Hilfe integrierter Mikrorechentechnik realisieren, durchaus sinnvoll verwendbar. Überdies ist er in der Literatur zum gängigen Terminus geworden. Die IMEKO als internationale Dachorganisation der Meßtechniker veranstaltet gar Symposien zum Thema „Intelligent Measurement". Daher meinen wir, bei aller Anerkennung der löblichen Absichten des Autors des obigen Beitrags, daß wir uns als zukunftsorientiertes Fachorgan dem Begriff der „intelligenten Meßtechnik" nicht verschließen können.

Zur Aktivierung des wissenschaftlichen Meinungsstreits zur KI in der Meßtechnik stellen wir den vom FA Metrologie und Standardisierung der WGMA vertretenen Standpunkt zur Diskussion. *Red. msr*

18. Weiterbildungskurse (Herbstkurse) TECHNISCHE KYBERNETIK

Vom 6. bis 10. Februar 1989 war Dresden Tagungsort für die o. g. Veranstaltung. Diese von der Kooperationsgemeinschaft „Technische Kybernetik" alljährlich veranstalteten und in diesem Jahr vom Wissenschaftsbereich Automatisierungstechnik der Sektion Informationstechnik der TU Dresden durchgeführten Kurse sind dem Weiterbildungsgedanken verpflichtet, was nun erstmals auch zur sinnvollen Ergänzung des aus organisatorischen Gründen mit der Jahreszeit häufig nicht korrespondierenden Namens „Herbstkurse" führte. Gerade auf den Weiterbildungsaspekt verwies Prof. Dr. sc. techn. H. Töpfer/TU Dresden in seinem einleitenden Grußwort mit der Bemerkung, daß die Kurse in erster Linie für Absolventen eines automatisierungstechnischen Studiums gedacht sind. Außerdem legte er Wert auf die Feststellung, daß die Themenwahl in diesem Jahr besonders auf das gerade von ihm nachhaltig vertretene Anliegen hinziele, das besagt, daß die Automatisierung kontinuierlicher Prozesse und die Fertigungsautomatisierung ein gemeinsames Dach haben, daß es gilt, die Gemeinsamkeiten herauszuarbeiten und das gegenseitige Verständnis zu fördern und somit Versäumnisse der Vergangenheit zu korrigieren.
Inhaltlich waren die Weiterbildungskurse in die folgenden 4 jeweils 16 Stunden umfassenden Themengruppen gegliedert:

Sektion 1
Prof. Dr.-Ing. habil. K. Reinschke/ZKI Dresden: „Sicherheit und Zuverlässigkeit von automatisierten Systemen"

Sektion 2
Prof. Dr. sc. nat. F. Baldeweg/ZfK Rossendorf, u. a.: „Prozeßüberwachung und Prozeßsicherung in technologischen Anlagen"

Sektion 3
Doz. Dr.-Ing. G. Reinig/TU Dresden, u. a.: „Prozeßleitsysteme"

Sektion 4
Prof. Dr. sc. techn. J. Schaller/TU Dresden: „Probleme der Fertigungsautomatisierung".

Dabei liefen stets 2 Sektionen (1 und 3 bzw. 2 und 4) parallel.
Großtechnische Havarien und Gefahrensituationen der zurückliegenden Jahre haben das Bewußtsein für Sicherheit und Zuverlässigkeit als technische Kategorien zweifellos nachhaltig beeinflußt. Unter diesem Blickwinkel traf die Themengruppe 1 ein hochaktuelles Thema. In geschliffener Vortragsweise bot K. Reinschke das entsprechende Theoriengebäude dar, arbeitete Anliegen und Aufgaben der Sicherheits- und Zuverlässigkeitstechnik heraus, definierte die entsprechenden Kenngrößen, gab auf der Basis der mathematischen Statistik einen Überblick über den experimentellen Zuverlässigkeitsnachweis, sprach zur Zuverlässigkeitsmodellierung und zu Redundanzprinzipien, erläuterte Boolesche Systemmodelle und Erweiterungen, Prozeßablaufbeschreibungen mit Zustandsübergangsgraphen und Markovschen Modellen sowie fehlertolerante Systeme und Wartungs- und Instandhaltungsstrategien zur Gewährleistung von Zuverlässigkeits- und Sicherheitsanforderungen.
In erster Linie am Beispiel von Kernkraftwerken war die 2. Themengruppe den relevanten Fragen der Technischen Diagnostik zur Prozeßüberwachung und -sicherung gewidmet. Dabei wurden das informationelle System besprochen und die mathematischen Beschreibungsmittel vorgestellt. Fragen der Mensch-Maschine-Kommunikation wurde große Aufmerksamkeit geschenkt. Um der in komplexen Anlagen großen Anzahl von Alarmen und den oft schwierigen Diagnosen erfolgreich begegnen zu können, wurde die Operatorberatung mit Hilfe von Expertensystemen vorgeschlagen. Dafür wurden die theoretischen Grundlagen des Problemlösens vermittelt und der grundsätzliche Aufbau von Expertensystemen angegeben. Einen hohen Stellenwert hatte dabei die Wissensdarstellung. Als Lösung mit Perspektive wurden schließlich Expertensysteme ohne Regelbasis, also künstliche neuronale Systeme, die sich an der Wissensverarbeitung in biologischen Systemen orientieren, erläutert, soweit sie bis heute erforscht sind.
Für Nutzer von Automatisierungsvorhaben bot speziell die Themengruppe 3 „Prozeßleitsysteme" eine Fülle von Informationen. Im Hinblick darauf, daß der Begriff „Prozeßautomatisierung" besonders auf die prozeßnahen Funktionen gerichtet ist, werden mit „Prozeßleiten" stärker die übergeordneten Funktionen betont. Somit wurde zunächst dem Hierarchiegedanken der gebührende Raum zuerkannt, und er wurde mit den komplexen Automatisierungsfunktionen verknüpft. Im Zusammenhang mit dem CIM-Referenzmodell wurden die Komponenten und Strukturen von Prozeßleitsystemen sowie die Niveaustufung und die adäquate Gerätetechnik besprochen. Speziell für die prozeßnahen Komponenten wurden auch Zuverlässigkeits- und Redundanzkonzepte erläutert. Die Mensch-Maschine-Kommunikation und entsprechende ergonomische Betrachtungen bildeten einen weiteren Schwerpunkt, da gerade diesbezügliche Maßnahmen für die Arbeit mit Prozeßleitsystemen von großer Bedeutung sind. Ein weiterer Akzent wurde mit einer sehr übersichtlichen Darstellung der Datenkommunikation (lokale Netze, Bussysteme, Datenzugriffsverfahren, OSI-Referenzmodell, MAP, PROWAY-C, Feldbuskonzepte) gesetzt. Mit Überblicksvorstellungen der DDR-Systeme audatec und S 2000 R wurden viele Teilnehmerwünsche befriedigt. Abschließende Bemerkungen dienten der Einordnung und der Weiterentwicklung von Prozeßleitsystemen und damit im Zusammenhang stehenden Problemen (Sprachentwicklung — Echtzeitsprachen, Funktionsumfang — „advanced control", Personalcomputer in der Prozeßautomatisierung, Bewertung von Prozeßleitsystemen usw.).
Probleme der Fertigungsautomatisierung, präziser die Fertigungsleittechnik für eine situationsabhängige Prozeßvorbereitung und -führung, standen im Mittelpunkt der Themengruppe 4. Sie bot das Rüstzeug für den schrittweisen Aufbau des flexiblen automatisierten, rechnerintegrierten Betriebes (CIM). Zielstellungen sind die flexible kundenwunschabhängige Konstruktion, die flexible situationsabhängige Prozeßvorbereitung und die flexible echtzeitüberwachte Produktion. Ohne die übergeordneten Ebenen (Steuerebene, Leitebene) zu vergessen, wurden vor allem Aussagen zu Fertigungsleitständen im prozeßnahen Bereich getroffen. Bestechend war in dieser Themengruppe die Praxisnähe, was sich vor allem in einem „Praktikum" der Teilnehmer (z. B. Programmierung einer numerisch gesteuerten Drehmaschine) in den Praktikumsräumen der Sektion 14 „Fertigungstechnik und Werkzeugmaschinen" der TU Dresden und in der Besichtigung eines Betriebes der flexiblen Fertigung ausdrückte. Damit wurden auch neue Maßstäbe für künftige Weiterbildungskurse gesetzt.

Zusammenfassend läßt sich sagen, daß die 5 Dresdener Tage eine äußerst effektive Form der Wissensvermittlung boten. Die mehr als 200 Teilnehmer werden sicher mit aufgefrischten und neuen Kenntnissen an ihre Tagesaufgaben zurückgekehrt sein. Als Gedankenstütze konnten sie neben ihren Mitschriften informatives Tagungsmaterial zu den Themengruppen 1, 3 und 4 mit nach Hause nehmen. Die Tagungsorganisation, für die Prof. Dr. sc. techn. R. Strietzel/TU Dresden verantwortlich zeichnete, ging mit der inhaltlichen Qualität der Weiterbildungskurse konform. Bleibt zu wünschen, daß in den folgenden Jahren noch mehr Teilnehmer die Chance der konzentrierten Kenntniserweiterung wahrnehmen.

msr 9057 D. Werner

11. Spezialistentagung „Mehrkriterielle Entscheidung“

Die Tagung, die vom 5. bis 9. Dezember 1988 in Breitenbrunn stattfand, wurde inhaltlich von der DDR-Arbeitsgruppe „Mehrkriterielle Entscheidung“ getragen. Der Leiter dieser Gruppe, Prof. J. Ester, leitete auch die Tagung, an der 37 Personen teilnahmen. Folgende Einrichtungen des Ministeriums für Hoch- und Fachschulwesen (MHF) waren beteiligt: TU Karl-Marx-Stadt, TU Dresden, TH Ilmenau, TH Leuna-Merseburg, Humboldt-Universität zu Berlin, Martin-Luther-Universität Halle, TH Leipzig.
Aus anderen Bereichen nahmen Vertreter folgender Institutionen teil: Chemische Werke Buna, Aprotex Limbach, Wäsche-Union Mittweida, Planeta Radebeul, Zentralinstitut für sozialistische Wirtschaftsführung, AdW der DDR, SV Dynamo.
Als ausländische Gäste nahmen Frau Dipl.-Ing. V. Marzen von der Ruhr-Universität Bochum und Prof. I. Stancev vom „Karl-Marx-Institut“, Sofia teil.
Der durch die Tagung organisierte wissenschaftliche Erfahrungsaustausch zwischen Mathematik, Kybernetik, Informatik und Anwendern war zum wiederholten Male außerordentlich fruchtbar.
Es wurde eine Vielzahl von Anregungen und Hinweisen aus der Praxis für weitere theoretische und methodologische Untersuchungen gegeben. Andererseits konnte die Theorie eine Reihe von Hinweisen zur Bewältigung praktischer Probleme beisteuern. Der Einfluß der Tagung wird in Planung, operativer Steuerung, Projektierung und Entwurf von Prozessen und Anlagen in folgenden Gebieten spürbar sein:

- Operative Steuerungen (6 Vorträge)
- Verfahrenstechnik (6 Vorträge)
- Standortoptimierung (3 Vorträge)
- Versorgungssysteme (1 Vortrag)
- Interaktive Entscheidungshilfen und theoretische Grundlagen (15 Vorträge).

Die offene, kritische und z. T. auch sehr ausführliche Diskussion zwischen Theoretikern und Praktikern führte dazu, daß der Erkenntniszuwachs für jeden Teilnehmer als sehr hoch eingeschätzt werden kann. Es bestätigte sich die Entwicklungstendenz der Schaffung von Entscheidungssoftware, die wichtige Moduln für CAD/CAM/CIM-Projekte darstellt. Gleichzeitig wurde die Verbindung mit Expertensystemen betont.
Erstmalig beteiligten sich Gesellschaftswissenschaftler mit Vorträgen und in der Diskussion substantiell an der Weiterentwicklung des Gebietes. Es wurde deutlich, daß bei den meisten praktischen Problemen technische, wirtschaftliche und soziale Effekte zu behandeln sind, was den interdisziplinären Charakter der Forschung auf dem Gebiet der Entscheidungstheorie- und -software nachdrücklich bestätigt.
Ergebnis der Tagung wird auch in diesem Jahr eine Reihe von Nachnutzungen wissenschaftlicher Ergebnisse in der Industrie oder in anderen Bereichen des MHF bzw. der AdW der DDR sein.
Aus theoretischer Sicht ergaben sich neue Erkenntnisse und neue Probleme hinsichtlich der sogenannten ε-Effizienz, der Überführung spezieller nichtlinearer mehrkriterieller Optimierungsaufgaben in duale (teilweise lineare) Ersatzaufgaben und der systematisierten Charakterisierung effizienter Lösungen durch entsprechende Optimalitätsbedingungen.
Bezüglich der interaktiven Entscheidungssysteme POLYP (TU Karl-Marx-Stadt), REH (ZKI der AdW, Berlin) und PROGRESS (ICT, Berlin) wurden die neuesten Entwicklungen vorgestellt bzw. demonstriert.
Initiiert durch die Vorträge von C. Dahme (HU Berlin) und I. Stancev („Karl-Marx-Institut“ Sofia) spielten in allen Diskussionen Fragen der wissenschaftlichen Methodologie der Systemanalyse eine wesentliche Rolle.
14 ausgewählte Vorträge werden in einem Heft der „Wissenschaftlichen Schriftenreihe der TU Karl-Marx-Stadt“ veröffentlicht. Die 12. Tagung ist für 1989 in Großbreitenbach vorgesehen.

msr 9034 J. Ester

9. ICAMC-Tagung 1988

Vom 3. bis 8. Oktober 1988 fand in Varna/VR Bulgarien die 9. ICAMC-Tagung (ICAMC Internationales Komitee für Automatisierung der Gruben) zum Thema „Automatisierung in den Gruben und Tagebauen“ statt. Die Tagung führte Wissenschaftler und Ingenieure aus zehn Ländern des Fachgebietes „Automatisierung des Bergbaus“ zusammen. In fünf Sektionen

- Grundfragen der Steuerung und Robotisierung in der Bergbauindustrie
- Systeme der automatischen Steuerung der Produktion in der Bergbauindustrie
- Technische Mittel und Systeme zur Automatisierung der Produktion in der Bergbauindustrie
- Automatisierung der Ingenieurarbeit in der Projektierung (CAD-Bergbau)
- Systeme und technische Mittel zur Erhöhung der technischen Sicherheit und Arbeitshygiene

wurden etwa 110 Vorträge gehalten.
L. Szklarski (VR Polen) wies in seinem Plenarvortrag auf die Notwendigkeit der Abschaffung schwerer Handarbeit im Bergbau hin. Dieses Ziel kann nur mit dem komplexen Einsatz der Mikroelektronik erreicht werden. Inhalt jetziger Forschungsaufgaben ist die Entwicklung von Automatisierungsstrategien zur komplexen Steuerung von Gruben und Schächten. Diese Zielstellung kam in den Vorträgen von A. Trop und G. Bagautinov (UdSSR) „Einschätzung des Automatisierungsniveaus des technologischen Prozesses bei der Entwicklung von automatischen Systemen zur Lenkung und Leitung der Produktion“ und I. Kukharenko u. a. (UdSSR) „Theoretische Aspekte der Dispatcherleitung eines untertägigen Schachtes“ zum Ausdruck.
Die Automatisierung von Bergbaumechanismen einschließlich der Technischen Diagnostik derartiger Systeme ist ausführlich diskutiert worden. Hierzu sind die Vorträge von J. Demoulin (Frankreich) „Automatisierung des Betriebes von Walzenschrämladern im Streb“ und W. A. Reznikow (UdSSR) „Systeme für Diagnose und Überwachung von Kennwerten der Bergbaumaschinen“ zu nennen.
Der Aufbau von Mikroprozessorsystemen und ihr Einsatz unter den besonderen Bedingungen des Bergbaus sind von E. Sarkow u. a. (VR Bulgarien) „Mikroprozessorsystem mit verteilter Architektur für die operative Zentralsteuerung eines Bergwerkes“ und Z. Petrusic und V. Nikolic (SFR Jugoslawien) „Autonome Testung von Mikrorechnersystemen in der Bergbauindustrie“ sowie weiteren Referenten vorgestellt worden.
International hat sich in den letzten Jahren die computergestützte Projektierung von Tagebauen, Gruben und technischen Ausrüstungen, wie Bewetterung und Wasserhaltung, sowie die rechnergestützte Bewertung von Lagerstätten enorm weiterentwickelt. Die Vorträge von W. A. Malyshew und I. I. Pomelnikow (UdSSR) „Bewertung der Bergwerkskapazität ausgehend von den Lagerstättenbedingungen mit Hilfe des Computerprojektierungssystems“, E. Koleva „Gestaltung des Böschungswinkels von Tagebauwänden und Strossen, die im homogenen Gebirge mit flacher Böschung aufgebaut sind, unter Einsatz von EDV“ sowie G. Dimitrov u. a. „Datenverarbeitungssystem für die Projektierung der Grubenbewetterung“ aus der VR Bulgarien sind dafür Beispiele.
In diesem Tagungsbericht konnte nur auf eine kleine Auswahl der Vorträge eingegangen werden. Das ausführliche Tagungsmaterial der 9. ICAMC-Tagung 1988 liegt in englischer und russischer Sprache beim Verfasser des Berichtes vor.

msr 9046 K.-P. Grober

Rauschen als Information. Von W. DENDA. Berlin: VEB Verlag Technik 1988. 188 Seiten, 118 Bilder. 19,— M.
Bestell-Nr.: 553 908 5

Das Eigenrauschen eines Halbleiterbauelementes gestattet eine Aussage über seine Güte, insbesondere über innere Mängel. Somit können durch Rauschanalyse sowohl der Herstellungsprozeß bez. technologischer Fehler überprüft als auch für die Anwendung wichtige Aussagen über die zu erwartende Lebensdauer der Bauelemente gemacht werden. Diese volkswirtschaftlich sehr attraktive Methode wird bis heute aufgrund mangelnder Kenntnisse kaum genutzt. Der Autor will mit der vorliegenden Broschüre diesem Mangel abhelfen und dem Ingenieur in der Praxis eine Grundlage für die Nutzung der Rauschanalyse zur Güte- und Eignungskontrolle vermitteln.
Die Broschüre stellt eine nützliche Bereicherung der Literatur zur Rauschdiagnostik dar und gibt einen guten Überblick über die grundsätzlichen Möglichkeiten zur Nutzung auf dem Halbleitergebiet sowie die notwendigen Meß- und Analyseverfahren und kann insbesondere Elektronikern im Gerätebau empfohlen werden. Sie ist leider nicht ohne Mängel. Bei der Darstellung der Grundlagen der Rauschanalyse, insbesondere bei der Spektraldichte, und bei ihrer Anwendung in anderen Gebieten gibt es merkliche Lücken und Fehler, wodurch ein verzerrtes Bild der Methode und ihrer Möglichkeiten entsteht. Sicher wäre der Autor gut beraten gewesen, wenn er sich auf das Halbleiterrauschen beschränkt hätte und dies durch Beispiele, wie sich typische Fehler im Rauschsignal abbilden oder wie man mit kommerziellen Geräten (z. B. dem SM-System vom VEB MKD) Messung und Analyse durchführen kann, erweitert hätte.

B 2423 P. LIEWERS

Regelungstechnik für Ingenieure. Von M. REUTER. 6., durchgesehene Auflage. Braunschweig/Wiesbaden: Friedr. Vieweg & Sohn Verlagsgesellschaft mbH 1988. 431 Seiten, 313 Bilder.

Das Buch zählt zu den regelungstechnischen Grundlagenlehrbüchern des Vieweg-Verlagsprogramms. Über die bisherigen sechs Auflagen hinweg ist es mehrfach überarbeitet und vor allem den aktuellen Erfordernissen angepaßt worden. Bei bewußtem Verzicht auf die Zustandsraumdarstellung präsentiert es sich als recht umfassende und aktuelle Darstellung der klassischen Regelungstechnik. Dabei werden kontinuierliche wie zeitdiskrete, lineare wie nichtlineare und unstetige Regelkreisglieder und Regelkreise gleichermaßen detailliert betrachtet, wobei die Beschreibung, die Stabilität und der Regelkreisentwurf im Mittelpunkt stehen. Sicher von besonderem Wert sind die kompakten und übersichtlichen Darstellungen von Analog- und digitaler Mikrorechentechnik und deren Verwendung zur Regelkreissimulation bzw. zur digitalen Regelung. Dank der geschickt gewählten Beispiele, häufig parallel aus den Bereichen Elektrotechnik bzw. Mechanik/Maschinenbau gewählt und damit viele ansprechend, und der Übungsaufgaben (mit Lösungen im Anhang) wird die Mitarbeit des Lesers didaktisch sinnvoll gefordert. Was das Buch aber wohl vor allem auszeichnet, ist die durchgängige Sorgfalt im Detail. Dafür verdienen Autor und Verlag hohes Lob. Als ingenieurmäßige Darstellung im besten Sinne macht das Buch seinem Titel alle Ehre.

B 2461 D. WERNER

Regelungssysteme im Zustandsraum II. Synthese. Von E. FREUND unter Mitwirkung von H. HOYER. München/Wien: R. Oldenbourg Verlag 1987. 223 Seiten, 16 Bilder.

Der durch den Band I „Struktur und Analyse" fundiert vorbereitete Band II „Synthese" widmet sich in didaktisch bestechender Weise den linearen kontinuierlichen Systemen mit Zustandsrückführungen bei Polvorgabe. Im Unterschied zu der Eindeutigkeit bei Eingrößensystemen ergibt sich dabei im Mehrgrößenfall eine gewisse Mehrdeutigkeit, auf die besonders aufmerksam gemacht wird.
Ein wichtiges Problem ist das der Entkopplung von Mehrgrößensystemen. Erstmalig für deutschsprachige Leser wird in diesem Zusammenhang die direkte Systembeschreibung und die damit verbundene Integratorentkopplung vorgestellt. Der Systementwurf über Entkopplung und Dynamikgestaltung wird ausführlich erläutert.
Der für kontinuierliche Steuerungen notwendige Beobachterentwurf folgt den Prinzipien des Reglerentwurfs. Bei der Polvorgabe für den Beobachter werden die bestehenden Analogien deutlich, u. a. die Mehrdeutigkeit im Mehrgrößenfall. Mit dem Separationsprinzip wird die Unabhängigkeit von Regler- und Beobachtersynthese nachgewiesen und am Beispiel diskutiert.
Alle Probleme sind kurz und leicht verständlich dargestellt. Wie im Band I vertiefen auch hier interessante Beispiele aus der Luft- und Raumfahrt, der Förder- und Antriebstechnik die abgeleitete Theorie. Die Beispiele und Übungsaufgaben sind durchgehend vorgerechnet und erhöhen das Verständnis weiter.
Der Klarheit und Exaktheit wegen ist das Buch Lehrenden und Lernenden gleichermaßen von Nutzen.

B 2369 M. RADTKE

Innovationsprozeß Flexible Automatisierung. Herausgegeben von H.-G. LAUENROTH. Berlin: Akademie-Verlag 1988. 388 Seiten. 24,— M.

Das Buch erschien als Band 25 innerhalb der Reihe Wissenschaft und Gesellschaft. 9 Autoren aus der AdW und dem Hochschulwesen der DDR lieferten Beiträge zum aktuellen Thema der flexiblen Automatisierung.
In den vergangenen 4 Jahrzehnten hat die Automatisierung in der Volkswirtschaft der DDR sowohl eine bemerkenswerte quantitative Entwicklung durchlaufen als auch einen tiefgreifenden qualitativen Wandlungsprozeß erfahren. Die zu Beginn der fünfziger Jahre eingeführte Einzweckautomatisierung erreichte in den siebziger Jahren eine vorläufige Sättigungsphase. Neue Bedürfnisse nach erweiterten Produktionssortimenten und kürzeren Innovationszyklen sowie neue Anforderungen zur Steigerung von Produktivität und Effektivität der gesellschaftlichen Produktion führten zur Entwicklung der flexiblen Automatisierung und gleichzeitig zum Konzept komplexer Lösungen in Form der computerintegrierten Fertigung (CIM-Konzept), das die Erneuerung und Rationalisierung geschlossener Reproduktionsprozesse in ihrer Gesamtheit umfaßt.
Die Beiträge behandeln, vom Herausgeber in die richtigen Proportionen gebracht, folgende Schwerpunkte, die recht lose aneinander gereiht sind:

— Computergestützte Systemanalyse des Innovationsprozesses flexible Automatisierung
— Einsatzstrategien und Effektivität flexibler Fertigungssysteme
— Analyse, Effektivität und Innovationsstrategien des Industrierobotereinsatzes
— Innovationsstrategien für CAD/CAM-Systeme
— Softwarestrategien für die flexible Automatisierung
— Forschungsstrategien für die 3. Robotergeneration
— Sozialökonomische Wirkungen und Innovationseffekte der flexiblen Automatisierung.

Mit statistischen Aussagen in tabellarischer Form und mit Skizzen wird der Text gut unterstützt.
Eine gelungene Arbeit im Grenzgebiet von Technik- und Gesellschaftswissenschaften!

B 2390 H. FUCHS

Kurzfassungen

HARTWIG, M.; STEIN, E.; STROBEL, R.:

Programmieren mit Ada

msr, Berlin 32 (1989) 6, S. 242—244

Auf dem in der DDR konsequent eingeschlagenen Weg in Richtung der Entwicklung und des Einsatzes von Hochtechnologien wird immer deutlicher, daß der Effektivität der Softwareproduktion eine entscheidende Bedeutung zukommt. Langfristig gesehen ist dieses Problem nur mit der Einführung moderner Softwaretechnologien zu lösen. In diesem Zusammenhang verdient die Programmiersprache Ada besondere Beachtung. Mit dem am ZKI der AdW entwickelten Ada-0-System besteht zum ersten Mal auch in der DDR die Möglichkeit der praktischen Erprobung wichtiger Konzepte dieser Sprache.

BALDEWEG, F.; BECHSTEIN, J.; FIEDLER, U.:

Zur Anwendung assoziativer Neuronennetze für die Diagnose und Therapiesteuerung

msr, Berlin 32 (1989) 6, S. 244—247

Der Trend in der Entwicklung von Systemen der Prozeßüberwachung und -sicherung komplexer technischer und technologischer Anlagen deutet auf zunehmende Integration echtzeitfähiger und wissensbezogen arbeitender informationeller Systeme hin. Die heutige Rechentechnik bietet für anspruchsvolle Probleme (z. B. Diagnose und Therapiesteuerung), bedingt durch die sequentielle Arbeitsweise, keine befriedigenden Lösungen. Der Beitrag hat Überlegungen und Vorstellungen zum Inhalt, die sich auf den Einsatz spezieller Hardware, z. B. assoziativ und parallel arbeitender Neuronennetze, für spezielle Aufgaben der Diagnose- und Therapiesteuerung beziehen.

BEICHELT, F.; SPROSS, L.:

Ein Verfahren zur approximativen Berechnung der Zuverlässigkeit komplizierter Systeme

msr, Berlin 32 (1989) 6, S. 248—251

Zur Berechnung von oberen und unteren Schranken für die Verfügbarkeit einer großen Klasse binärer monotoner Systeme wird ein Verfahren vorgestellt, das im Unterschied zu bekannten Abschätzungen nur einen Bruchteil der minimalen Pfad- und Schnittmengen des Systems benötigt. Das Verfahren liefert auch für größere Systeme nach relativ kurzen Rechenzeiten Schranken ausreichender Genauigkeit.

GRUNDIG, C.-G.:

Zur Methodik der experimentell-analytischen Ermittlung von Gesetzmäßigkeiten diskreter Fertigungsprozesse

msr, Berlin 32 (1989) 6, S. 251—256

Grundlage für eine gezielte Gestaltung und Steuerung diskreter Fertigungsprozesse sind gesicherte Kenntnisse über deren innere Wirkungszusammenhänge. Der zugehörige Erkenntniszustand ist unbefriedigend. Im Beitrag werden methodische Schritte einer speziellen Analysestrategie der Versuchsplanung und -auswertung entwickelt, bei der in Verbindung mit gezielt aufgebauten Simulationsexperimenten eine mathematisch-analytische Modellierung der Wirkungszusammenhänge vorgenommen werden kann.

ROTHE, L.:

Optimierungsalgorithmen der Aperturbelegung höchstfrequenter Strahlungssensoren

msr, Berlin 32 (1989) 6, S. 262—266

Mikrowellensensorsysteme mit phasen- und amplitudengleicher Anregung weisen eine systembedingt abhängige Fehlanpassung an den Feldraum des realen Strahlungsfeldes auf. Zur Ermittlung möglicher Anpassungskonzeptionen von Sensor- und Feldcharakteristik werden auf der Grundlage adaptiver Mechanismen Systemlösungen diskutiert. Ausgangspunkt des Optimierungsprozesses ist die amplituden- und phasenabhängige Zustandskonfiguration der Aperturbelegung des Strahlungssensors.

ZIPSER, L.; LABUDE, J.:

Planarer akustischer Abluftfeuchtesensor

msr, Berlin 32 (1989) 6, S. 267—270

Vorgestellt wird ein neuer Gasfeuchtesensor auf der Basis fluidisch-akustischer Oszillatoren. Der Sensor ist für den Einsatz unter extremen Prozeßbedingungen, z. B. zur Messung des Feuchtegrades der heißen und verschmutzten Abluft industrieller Trocknungsanlagen, geeignet. Mit dem Sensor können Trocknungsprozesse energieoptimal gestaltet und erhebliche Energiemengen eingespart werden.

Резюме

HARTWIG, M.; STEIN, E.; STROBEL, R.:

Программирование при помощи АДА

msr, Berlin 32 (1989) 6, стр. 242—244

По пути, по которому ГДР настойчиво идет по направлению развития и применения высоких технологий все более очевидно, что эффективность производства математического обеспечения приобретает решающее значение. На более длительный период данная проблема решается лишь путем введения современных технологий матобеспечения. В этой связи язык программирования АДА получает особое внимание. При помощи системы АДА-О, разработанной в Центральном институте кибернетики и информационных процессов при Академии наук, впервые также и в ГДР имеется возможность практического испытания важных концептов этого языка.

BALDEWEG, F.; BECHSTEIN, J.; FIEDLER, U.:

О применении ассоциативных сетей неуронов для диагноза и управления терапией

msr, Berlin 32 (1989) 6, стр. 244—247

Тенденция в области развития систем контроля и сохранения процесса комплексных технических и технологических установок показывает на возрастающую интеграцию информационных систем, работающих в реальном масштабе времени и по отношению к знаниям. Современная вычислительная техника для сложных проблем (напр. диагноза и управления терапией), обусловлено последовательным режимом работы, обеспечивает никаких удовлетворительных решений. В данной статье указаны соображения и представления, относящиеся к применению специального аппаратурного обеспечения, напр. сетей неуронов, работающих ассоциативным и параллельным образом, для целей специальных задач управления диагнозом и терапией.

BEICHELT, F.; SPROSS, L.:

Метод аппроксиматиного расчета надежности сложных систем

msr, Berlin 32 (1989) 6, стр. 248—251

Для расчета верхних и нижних барьеров готовности большого класса двоичных монотонных систем представлен метод, который в отличие от известных оценок требует лишь небольшой части минимальных количеств трактов и интерфейсов системы. Метод дает и для более высоких систем после относительно короткого времени расчета барьеры достаточной точности.

GRUNDIG, C.-G.:

О методике экспериментально-аналитического определения закономерностей дискретных процессов изготовления

msr, Berlin 32 (1989) 6, стр. 251—256

Основой устремленного оформления и управления дискретными процессами изготовления являются точные знания о их внутренних связях действия. Соответствующее состояние знаний является неудовлетворительным. В статье указаны методические шаги специальной стратегии анализа планирования и оценки опытов, при которой в связи с устремленно построенными экспериментами моделирования может быть проведено математически-аналитическое моделирование связей действия.

ROTHE, L.:

Алгоритмы оптимирования выбора отверстия высокочастотных датчиков излучения

msr, Berlin 32 (1989) 6, стр. 262—266

Системы с микроволнистыми датчиками с активизацией, равной фазам и амплитудам показывают зависящую обусловлено системой ошибочную адаптацию к пространству поля реального поля излучения. Для определения возможных концепций адаптации характеристики датчика и поля на основе адаптивных механизмов обсуждаются системные решения. Исходным пунктом процесса оптимирования является конфигурация состояния, зависящая от амплитуды и фазы, определения отверстия датчика излучения.

ZIPSER, L.; LABUDE, J.:

Планарный акустический датчик влажности отработанного воздуха

msr, Berlin 32 (1989) 6, стр. 267—270

Представлен новый датчик влажности газа на основе флюидноакустических осцилляторов. Датчик годится для применения в крайних условиях процесса, напр. для измерения степени влажности горячего и запыленного отработанного воздуха промышленных сушильных установок. При помощи датчика возможно оптимальное с точки зрения энергии оформление сушильных процессов, а также возможна экономия значительного количества энергии.

Summaries

Résumés

HARTWIG, M.; STEIN, E.; STROBEL, R.:

Programming in Ada

msr, Berlin 32 (1989) 6, pp. 242—244

Due to the increasing use of computers the efficiency of software production becomes more and more important. For this reason modern software technologies are to be introduced. Therefore the programming language Ada should be taken particularly into consideration. By means of the Ada-O system developed by the ZKI of the Academy of Sciences there is for the first time in GDR the possibility for practical tests of important concepts of this language.

BALDEWEG, F.; BECHSTEIN, J.; FIEDLER, U.:

On the Application of Associative Neuron Networks for Diagnosis and Therapy Control

msr, Berlin 32 (1989) 6, pp. 244—247

The trend in the development of process supervision and safety systems for complex technical and technological plants shows an increasing integration of real-time and knowledge-based information systems. For pretentious problems, such as diagnosis and therapy control, the present computing does not offer satisfying solutions due to its sequential way of working. The authors present considerations and ideas related to the use of special hardware, e.g. neuron networks functioning in an associative and parallel way, for specific problems of diagnosis and therapy control.

BEICHELT, F.; SPROSS, L.:

A Method for Approximative Calculation of the Reliability of Complicated Systems

msr, Berlin 32 (1989) 6, pp. 248—251

For the calculation of upper and lower limits for the availability of a large class of binary monotonous systems, the authors present a method which only needs a fraction of the minimal path and intersection sets of the system in contrast to known estimations. For larger systems, too, the method provides limits of sufficient accuracy after a relatively short computing time.

GRUNDIG, C.-G.:

On the Methodism of the Experimentally Analytical Determination of Laws of Discrete Manufacturing Processes

msr, Berlin 32 (1989) 6, pp. 251—256

The basis for a systematical design and control of discrete manufacturing processes is a founded knowledge on their inner causal relations. The corresponding state of knowledge is not satisfying. The author develops methodical steps of a special analysis strategy of the experiment planning and evaluation which allow a mathematically analytical modeling of the causal relations in connection with simulation experiments systematically designed.

ROTHE, L.:

Optimization Algorithms for the Aperture Coverage of Very High Frequency Radiation Sensors

msr, Berlin 32 (1989) 6, pp. 262—266

Microwave sensor systems with equal phase and amplitude excitation show a system-dependent mismatch to the field space of the real radiation field. For the determination of possible matching conceptions of sensor and field characteristic, the author discusses solutions based on adaptive mechanisms. The starting point of the optimization process is the amplitude- and phase-dependent state configuration of the aperture coverage of the radiation sensor.

ZIPSER, L.; LABUDE, J.:

Planar Acoustical Waste Air Humidity Sensor

msr, Berlin 32 (1989) 6, pp. 267—270

The authors present a new gas humidity sensor based on fluid-acoustical oscillators. The sensor is suitable for application under extreme process conditions, e.g. for the measurement of the humidity degree of the hot and polluted exhaust air of industrial drying plants. By the use of the sensor described the drying processes can be realized in an energy-optimal way, thus a considerable amount of energy being saved.

HARTWIG, M.; STEIN, E.; STROBEL, R.:

La programmation en Ada

msr, Berlin 32 (1989) 6, pp. 242—244

En conséquence de l'application accroissante des ordinateurs, l'efficacité de la production du logiciel devient de plus en plus important. C'est pourquoi, il faut introduire de modernes technologies de logiciel. A cet égard, on doit tenir compte en particulier du langage de programmation Ada. A l'aide du système Ada-O développé par le ZKI de l'académie des sciences, pour la première fois il y a la possibilité en RDA d'éprouver pratiquement des concepts importants de ce langage.

BALDEWEG, F.; BECHSTEIN, J.; FIEDLER, U.:

Sur l'emploi des réseaux associatifs de neurons pour la commande de diagnostic et de thérapie

msr, Berlin 32 (1989) 6, pp. 244—247

La tendance du développement des systèmes de surveillance et de sécurité des installations techniques et technologiques complexes montre une intégration accroissante des systèmes d'information travaillant en temps réel et étant basés sur connaissance. Pour les problèmes exigeants (p.e. la commande de diagnostic et de thérapie), les ordinateurs présents n'offrent pas des solutions satisfaisantes en conséquence de leur mode séquentiel de fonctionnement. Les auteurs présentent des considérations et des idées qui se rapportent à l'emploi du matériel spécial, p.e. des réseaux à neurons travaillant d'une manière associative et parallèle, pour les problèmes spécifiques de la commande de diagnostic et de thérapie.

BEICHELT, F.; SPROSS, L.:

Une méthode pour le calcul approché de la fiabilité de systèmes compliqués

msr, Berlin 32 (1989) 6, pp. 248—251

Pour le calcul des limites supérieures et inférieures de la disponibilité d'une grande classe de systèmes monotones binaires, l'auteur présente une méthode qui contrairement aux estimations connues n'exige qu'une petite partie des ensembles de chemin et d'intersection du système. Pour les systèmes plus grands, la méthode fournit également les limites à une précision suffisante après un temps de calcul relativement court.

GRUNDIG, C.-G.:

Sur la méthodologie de la détermination expérimentalement analytique des lois des processus de fabrication discrets

msr, Berlin 32 (1989) 6, pp. 251—256

La base le projet et la commande systématiques des processus de fabrication discrets est une connaissance solide sur leurs relations causales intérieures. Le correspondant état de connaissance n'est pas satisfaisant. L'auteur développe des pas méthodiques d'une stratégie d'analyse spéciale du planning et de l'évaluation d'expériences qui permettent une modélisation mathématiquement analytique des relations causales en connexion avec des expériences de simulation systématiquement construites.

ROTHE, L.:

Algorithmes d'optimalisation pour la couverture d'ouverture de capteurs de rayonnement à très hautes fréquences

msr, Berlin 32 (1989) 6, pp. 262—266

Les systèmes de capteur à micro-ondes avec une excitation égale en phase et en amplitude montrent une désadaptation dépendante du système à l'espace de champ du champ réel de rayonnement. Pour la détermination des conceptions d'adaptation possibles des caractéristiques du capteur et du champ, l'auteur discute des solutions basées sur des mécanismes adaptatifs. Le point de départ du processus d'optimalisation est la configuration d'état dépendante d'amplitude et de phase de la couverture d'ouverture du capteur de rayonnement.

ZIPSER, L.; LABUDE, J.:

Capteur acoustique planaire pour l'humidité d'air vicié

msr, Berlin 32 (1989) 6, pp. 267—270

Les auteurs présentent un nouveau capteur d'humidité de gaz qui est basé sur des oscillateurs fluide-acoustiques. Le capteur est apte à l'emploi sous les conditions de processus extrêmes, p.e. pour la mesure du degré d'humidité de l'air vicié chaud des installations de séchage industrielles. A l'aide du capteur décrit, on peut réaliser les procédés de séchage d'une manière optimale en énergie et, en conséquence, économiser des quantités d'énergie considérables.

Neue Bücher aus der UdSSR

Hierbei handelt es sich um neue Titel aus der UdSSR, die dem sowjetischen Neuerscheinungsdient „Novye knigi" entnommen sind. Die Bücher können über den Volksbuchhandel bestellt werden.

[1] Informacionnye technologii v ispytanijach složnych ob'ektov: metody i sredstva (Informationstechnologien bei der Erprobung komplizierter Objekte: Methoden und Mittel). Naukova dumka (USSR). 2. Quart. 1990. NK 88-45 (49).

[2] Koševoj, V.: Vosstanovlenie signalov v akustičeskich informacionnych sistemach (Signalerkennung in akustischen Informationssystemen). Naukova dumka (USSR). 2. Quart. 1990. NK 88-45 (50).

[3] Grunskij, I.; Kozlovskij, V.; Ponomarenko, G.: Predstavlenie konečnych avtomatov (Darstellung endlicher Automaten). Naukova dumka (USSR). 2. Quart. 1990. NK 88-45 (54).

[4] Potapov, V.: Organizacionno-technologičeskoe proektirovanie v SAPR (Organisatorisch-technologische Projektierung in CAD). Technika (USSR). 3. Quart. 1990. NK 88-45 (117).

[5] Gostev, V.: Sistemy upravlenija s cifrovymi reguljatorami (Systeme zur Steuerung mit digitalen Reglern). Technika (USSR). 2. Quart. 1990. NK 88-45 (133).

[6] Kuznecov, B.; Sergeev, V.; Černyšev, V.: Mikroprocessornoe upravlenie mnogokanal'nymi sistemami vysokoj točnosti (Mikroprozessorsteuerung in mehrkanaligen Systemen hoher Genauigkeit). Technika (USSR). 4. Quart. 1990. NK 88-45 (134).

[7] Apokin, I.; Majstrov, L.: Istorija vyčislitel'noj techniki (Geschichte der Rechentechnik). Nauka 4. Quart. 1989. NK 88-45 (136).

[8] Kasatkin, V.; Zagljadnov, I.: Na ekrane EVM (Auf dem Bildschirm der EDV-Anlagen). Technika (USSR). 2. Quart. 1990. NK 88-45 (138).

[9] Višenčuk, I.; Čerkasskij, N.: Algoritmičeskie operacionnye ustrojstva i superEVM (Algorithmische Operationssysteme und Super-EDV-Anlagen). Technika (USSR). 2. Quart. 1990. NK 88-45 (142).

[10] Semenjuk, E.: Informatika: dostiženija, perspektivy, vozmožnosti (Informatik: Errungenschaften, Perspektiven, Möglichkeiten). Nauka. 4. Quart. 1989. NK 88-45 (218).

[11] Bryskin, V.; Akovlev, V.: Matematičeskie modeli programmno-celevogo planirovanija techničeskich sredstv (Mathematische Modelle der Zielprogramm-planung technischer Mittel). Nauka. 3. Quart. 1989. NK 88-46 (9).

[12] Ljung, L.: Identifikacija sistem. Teorija dlja pol'zovatelej (Identifikation von Systemen. Theorie für den Anwender. Übers. aus dem Engl.). 4. Quart. 1989. NK 88-46 (10).

[13] Trochimenko, Ja.: Vyčislenija na mikro EVM (Berechnungen auf dem Mikrorechner). Technika (USSR). 4. Quart. 1990. NK 88-46 (52).

[14] Avtomatizacija proektirovanija v elektronike (Automatisierung der Projektierung in der Elektronik). Ausg. 41. Technika (USSR) 2. Quart. 1990. NK 88-46 (71); Ausg. 42. Technika (USSR). 4. Quart. 1990. NK 88-46 (72).

[15] Adaptivnye sistemy avtomatičeskogo upravlenija (Adaptive Steuerungssysteme. Ausg. 18). Technika (USSR). 2. Quart. 1990. NK 88-46 (74).

[16] Ogarkov, M.: Metody statističeskogo ocenivanija parametrov slučajnych processov (Methoden der statistischen Schätzung von Parametern stochastischer Prozesse). Energoatomizdat. 4. Quart. 1989. NK 88-48 (47).

[17] Bessonov, A.; Zagašvili, Ju.; Markelov, A.: Metody i sredstva identifikacii dinamičeskich ob'ektov (Methoden und Mittel zur Identifikation dynamischer Objekte). Energoatomizdat. 2. Quart. 1989. NK 88-48 (114).

[18] Malyšev, N.; Micuk, N.: Osnovy optimal'nogo upravlenija processami avtomatizirovannogo proektirovanija (Grundlagen der optimalen Steuerung von Prozessen der automatisierten Projektierung). Energoatomizdat. 3. Quart. 1989. NK 88-48 (116).

[19] Proektirovanie sistem avtomatizacii technologičeskich processov (Projektierung von Systemen zur Automatisierung technologischer Prozesse). Energoatomizdat. 3. Quart. 1989. NK 88-48 (117).

[20] Veršinin, O.: Komp'jutery v sistemach upravlenija i kontrolja (Rechner in Systemen zur Steuerung und Kontrolle). Energoatomizdat. 1. Quart. 1989. NK 88-48 (121).

[21] Galanskij, B.; Poljakov, V.: Informacionnye sistemy. (Informationssysteme). Izd-vo Tomsk. un-ta. 2. Quart. 1989. NK 88-48 (198).

[22] Korikov, A.: Osnovy sistemnogo analiza i teorii sistem (Grundlagen der Systemanalyse und Systemtheorie). Izd-vo Tomsk. un-ta. 2. Quart. 1989. NK 88-48 (199).

[23] Kibernetika i vyčislitel'naja technika. Vyp. 85 (Kybernetik und Rechentechnik. Ausg. 85). Naukova dumka (USSR). 1. Quart. 1990. NK 88-49 (25).

[24] Teoretičeskaja kibernetika (Theoretische Kybernetik). Mecniereba (GruzSSR). 3. Quart. 1989. NK 88-49 (26).

[25] Jazykovye processory i raspoznavanie reči (Sprachprozessoren und Spracherkennung). Mecniereba (GruzSSR). 3. Quart. 1989. NK 88-49 (27).

[26] Kolpaev, V.; Frenkel', S.; Cygel'nyj, I.: Molekularnaja kibernetika (Molekularkybernetik). Izd-vo Lvov. un-ta (USSR). 1. Quart. 1990. NK 88-49 (49).

[27] Cyladze, M.; Dolidze, A.: Sistemy upravlenija bazami dannych, sostojanie i perspektivy razvitija (Systeme zur Steuerung von Datenbasen, Stand und Perspektiven der Entwicklung). Mecniereba (GruzSSR). 3. Quart. 1989. NK 88-49 (55).

[28] Teorija i ustrojstv sistem avtomatičeskogo upravlenija (Theorie und Anlagen von Systemen der automatischen Steuerung). Mecniereba (GruzSSR). 4. Quart. 1989. NK 88-49 (58).

[29] Cegelik, G.: Sistemy raspredelennych baz dannych (Systeme verteilter Datenbanken). Izd-vo Lvov. un-ta. 3. Quart. 1990. NK 88-49 (59).

[30] Avtomatizacija proizvodstvennych processov v mašinostroenii i priborostroenii (Automatisierung von Produktionsprozessen im Maschinenbau und Gerätebau). Izd-vo Lvov. un-ta (USSR). 3. Quart. 1990. NK 89-49 (72).

[31] Dinamičeskie sistemy. Vyp. 9 (Dynamische Systeme. Ausg. 9). Izd-vo Kiev. un-ta (USSR). 2. Quart. 1989. NK 88-49 (89).

[32] Issledovanie operacij i ASU (Operationsforschung und ASU). Ausg. 35. Izd-vo Kiev. un-ta (USSR). 1. Quart. 1990. NK 88-49 (94); Ausg. 36. Izd-vo Kiev. un-ta (USSR). 3. Quart. 1990. NK 88-49 (95).

msr 9050

Impressum

messen · steuern · regeln

Herausgeber: Kammer der Technik, Wissenschaftlich-Technische Gesellschaft für Meß- und Automatisierungstechnik (WGMA)

Verlag: VEB Verlag Technik, Oranienburger Str. 13/14, PSF 201, Berlin, DDR-1020, Telegrammadresse: Technikverlag Berlin; Telefon 2 87 00; Telex: 011 222 8 techn dd

Verlagsdirektor: Dipl.-Ing. Klaus Hieronimus

Redaktion: Dr.-Ing. Dietrich Werner, Verantwortlicher Redakteur (Telefon: 2 87 03 62)

Dr.-Ing. Jürgen Wede, Redakteur (Telefon: 2 87 03 74)

Gestaltung: Veronika Becker

Lizenz-Nr.: 1112 des Presseamtes beim Vorsitzenden des Ministerrates der Deutschen Demokratischen Republik

AN (EDV): 15 936

Erscheinungsweise: monatlich 1 Heft

Heftpreis: 4,— M, Abonnementpreis vierteljährlich 12,— M; Auslandspreise sind den Zeitschriftenkatalogen des Außenhandelsbetriebs BUCHEXPORT zu entnehmen.

Gesamtherstellung: VEB Druckerei „Thomas Müntzer", Bad Langensalza, 5820

Anzeigenannahme: Für Bevölkerungsanzeigen alle Anzeigen-Annahmestellen in der DDR, für Wirtschaftsanzeigen der VEB Verlag Technik, Oranienburger Str. 13/14, PSF 201, Berlin, 1020. Anzeigenpreisliste Nr. 7; Auslandsanzeigen: Interwerbung GmbH, Hermann-Duncker-Str. 89, Berlin, DDR-1157

Bezugsmöglichkeiten:

DDR: sämtliche Postämter

SVR Albanien: Direktorije Quendrore e Perhapjes dhe Propaganditit te Librit Bruga Konference e Pezes, Tirana

VR Bulgarien: Direkzia R.E.P., 11a, Rue Paris, Sofia

VR China: China National Publications Import and Export Corporation, West Europe Department, P.O. Box 88, Beijing

ČSSR: PNS — Ústřední Expedicia a Dovoz Tisku Praha, Slezska 11, 12000 Praha 2; PNS, Ústredná Expedicia a Dovoz Tlače, Posta 002 885 47 Bratislava

SFR Jugoslawien: Jugoslovenska Knjiga, Terazije 27, Beograd; Izdavačko Knjižarako Produzeče MLADOST, Ilica 30, Zagreb

Koreanische DVR: CHULPANMUL Korea Publications Export & Import Corporation, Pyongyang

Republik Kuba: Empresa de Comercio Exterior de Publicaciones, O'Reilly No. 407, Ciudad Habana

VR Polen: C. K. P. i. W. Ruch, Towarowa 28, 00-958 Warszawa

SR Rumänien: D.E.P. București, Piața Scienteii, București

Ungarische VR: P.K.H.I., Külföldi Előfizetési Osztály, P.O. Box 16, 1426 Budapest

UdSSR: Städtische Abteilungen von Sojuzpečat oder Postämter und Postkontore

SR Vietnam: XUNHASABA, 32, Hai Ba Trung, Hanoi

BRD und Berlin (West): ESKABE Kommissions-Grossobuchhandlung, Postfach 36, 8222 Ruhpolding/Obb.; Helios Literatur-Vertriebs-GmbH, Eichborndamm 141—167, Berlin (West) 52; Kunst und Wissen Erich Bieber OHG, Postfach 46, 7000 Stuttgart 1; Gebrüder Petermann, BUCH + ZEITUNG INTERNATIONAL, Kurfürstenstraße 111, Berlin (West) 30

Österreich: Helios Literatur-Vertriebs-GmbH & Co. KG, Industriestraße B 13, A-2345 Brunn am Gebirge

Schweiz: Verlagsauslieferung Wissenschaft der Freihofer AG, Weinbergstraße 109, 8033 Zürich

Alle anderen Länder: örtlicher Buchhandel; BUCHEXPORT Volkseigener Außenhandelsbetrieb der Deutschen Demokratischen Republik, Postfach 160, Leipzig, DDR-7010 und Leipzig Book Service, Talstraße 29, Leipzig, DDR-7010.

Abnahmebedingungen:

Die Redaktion veröffentlicht nur Originalbeiträge. Für Nachdrucke, Übersetzungen oder Vervielfältigungen jeglicher Art gilt der Rechtsvorbehalt. Die Redaktion behält sich Änderungsvorschläge bei der Beitragsbearbeitung vor. Die Beiträge müssen generell mit einer Veröffentlichungsfreigabe des Betriebes oder der Institution der Autoren versehen sein.

Bei der Anfertigung von Beiträgen sind unbedingt die Manuskripthinweise zu beachten. Diese sind dem Heft 1/1989, S. 45, zu entnehmen bzw. werden von der Redaktion auf Anfrage zugesandt. Die Einhaltung der Manuskripthinweise bildet die Voraussetzung für eine zügige und sachgerechte Beitragsbearbeitung.

6. Wissenschaftliche Konferenz der IH Zwickau

In der Zeit vom 21. bis 30. November 1989 führt die Ingenieurhochschule Zwickau ihre 6. Wissenschaftliche Konferenz unter dem Titel „Rationalisierung im Maschinenbau durch Schlüsseltechnologien" durch. In 9 Fachtagungen wird auf aktuelle Probleme, wie Applikation von Maschinenbaukeramik, CAD/CAM im Maschinen- und Fahrzeugbau, Umformtechnik, Handhabetechnik, Montagetechnik, Eigenspannungen und Oberflächenverfestigung, Elektroautomatisierungstechnik und Leitung des Produktionshauptprozesses, eingegangen.
Aus dem Tagungsangebot sei die Fachtagung 8 „Elektroautomatisierungstechnik" (4. Fachtagung Elektroautomatisierungstechnik) besonders hervorgehoben.
Schwerpunkte dieser Fachtagung sind:

Fachgruppe 1: Sensor- und Meßtechnik sowie Querschnittsprobleme der Elektroautomatisierungstechnik

- Meßtheorie, Meßinformationsgewinnung, -speicherung und -verarbeitung
- Sensoren und Sensorapplikationen, intelligente Sensorsysteme
- Mikro-, Einchip-, Kleinrechner- und PC-Anwendung zur Meßinformationsspeicherung und -verarbeitung, Steuerung und Automatisierung von Meßabläufen
- elektromagnetische Verträglichkeit und Schutz von Elektroautomatisierungsanlagen, Zuverlässigkeit und Verfügbarkeit

Fachgruppe 2: Antriebs- und Stelltechnik sowie komplexe Lösungen der Prozeß- und Elektroautomatisierung

- Steuerung und Positionierung von Gleichstrom-Stellantrieben bei hochdynamischer Beanspruchung
- optimale thermische Auslegung von Gleichstrom-Stellantrieben
- spezielle Probleme bei der Steuerung von Industrierobotern, Anpassung kommerzieller Steuerungen an konkrete Einsatzfälle
- komplexe Erzeugnisautomatisierung, dargestellt an Problemlösungen für Wärmepumpenanlagen und Elektronikausrüstungen für Kraftfahrzeuge.

Veranstaltungstermin der Fachtagung 8: 28. und 29. November 1989.

Weitere Auskünfte, auch zu den anderen Fachtagungen, erteilt:

Ingenieurhochschule Zwickau
Prorektor für Naturwissenschaften und Technik, Prof. Dr. sc. techn. R. Frobin
Dr.-Friedrichs-Ring 2A
Zwickau
9540
Tel.: Zwickau 82 32 34

IFAC Workshop System Structure and Control: State-space and Polynomial Methods

Vom 25. bis 27. September 1989 findet in Prag der o. g. Workshop statt. Ziel ist die Diskussion über die Vorteile der algebraischen und der geometrischen Methoden in der Systemtheorie und der Steuerung.

Schwerpunkte:

- Lineare Systeme
- Algebraische und geometrische Methoden
- Invarianten und Struktur
- Singuläre Systeme
- Zeitvariante Systeme
- Systeme unendlicher Dimension
- n-D-Systeme
- Zeitdiskrete Systeme
- Lineare Steuerstrategien
- Feedback and Feedforward
- Stabilität und Stabilisierung
- Robuste Regelung
- Parametrisierung von Reglern
- Polynomialgleichungen
- Numerische Algorithmen.

Korrespondenz:
IFAC Workshop
Institute of Information Theory
and Automation
18208 Prague 8
Czechoslovakia

9. Aachener Fluidtechnisches Kolloquium (AFK)

Vom 20. bis 22. März 1990 veranstaltet das Institut für hydraulische und pneumatische Antriebe und Steuerungen der Rheinisch-Westfälischen Technischen Hochschule Aachen das 9. AFK.
Die anwendungsorientierte Tagung wird von einer Ausstellung begleitet, in der Exponate aus den Bereichen Hydraulikzubehör, Meßtechnik, digitale Signalverarbeitung vorgestellt werden.
Tagungsthemen:

1. Betrieb und Instandhaltung hydraulischer Anlagen
2. Entwicklungen in der KFZ-Hydraulik
   - Anforderungen der Automobilhersteller
   - Lenkung, Bremse, Fahrwerk
   - Zentralhydraulik
   - Druckmedien
   - Motor- und Antriebshydraulik
3. Trends in der Ventiltechnik
   - Strömungsmechanische Grundlagen
   - Proportionalventile, Ansteuerelektronik
   - Schnellschaltventile
   - Low-Cost-Konzepte
   - Kompensation von Störeinflüssen
4. Erschließung neuer Anwendungsfelder durch moderne hydraulische Systemtechnik
   - Pumpen- und Motorregelung
   - Übertragung großer Leistung
   - Vorausbestimmung der Systemeigenschaften
   - Digitaler Signalkreis
   - Reglerauslegung
   - Sensorik
5. Neuere Entwicklungen pneumatischer Komponenten
   - Miniaturisierung
   - Stetigventile
   - Dichtungstechnik
   - Einsatz neuer Materialien
   - Ölfreier Betrieb
6. Moderne Anwendungen pneumatischer Systeme
   - Handhabungs- und Montagetechnik
   - Meß- und Prüftechnik
   - KFZ-Pneumatik
   - Integrierte Systeme

Korrespondenz:
Institut für hydraulische und pneumatische Antriebe und Steuerungen der RWTH Aachen
Steinbachstraße 53
D-5100 Aachen

International Symposium „Application of Multivariable System Techniques" (AMST 90)

Das „Institute of Measurement and Control" führt vom 18. bis 19. April 1990 an der University of Bradford sein 4. Symposium zur Anwendung von Techniken multivariabler Systeme durch.
Zu den Schwerpunkten gehören:

- Entwurfs-, Identifikations- und Implementierungsverfahren, Anwendungsprobleme
- Anwendung von multivariablen, selbstabstimmenden, robusten und adaptiven Steuerungs- bzw. Regelungsstrategien, Rechnersteuerung von Multiinput-/Multioutput-Prozessen
- Anwendung von multivariablen und Mehrebenensystemtechniken auf Robotermechanismen
- Fortgeschrittene CAD-Methoden für Steuerung und Simulation diskreter, kontinuierlicher, linearer oder nichtlinearer Systeme
- Vergleichende Fehlersicherungs- und Integritätsstudien verschiedener Ansätze für die Regelung von Mehrgrößensystemen.

Korrespondenz:
Continuing Education Unit
University of Bradford
Bradford
West Yorkshire BD71DP
U.K.

ISSN 0026-0347, mess., steuern, regeln
Berlin 32 (1989) 6, S. 241—288